



Системный администратор

ежемесячный журнал www.samag.ru

№6(139)
июнь 2014



Облако от VMware
Разворачиваем vCloud Director

Инфокиоск своими руками
Подготовка «общественного» ПК

Red Hat Network
Изучаем Katello, Pulp и Candlepin

USB_ModeSwitch
USB-сотовый модем и ОС Linux



Большой Брат видит тебя!

Скрытое наблюдение за пользователем

16+

Пятая пара

Бесполезные занятия

Проблемы российского ИТ-образования.
Взгляд изнутри. Предложения
по его совершенствованию

Мониторинг

Compliance Settings в SCCM 2012

Оценка и отслеживание достоверности
конфигураций компьютеров компаний,
внесение в них некоторых исправлений

Инструменты

Программирование USB в Android

Организация обмена информацией
между электронными устройствами
и аппаратами под управлением Android

МФУ

Panasonic

ПРОИЗВОДИТЕЛЬНОСТЬ

ЭКОНОМИЧНОСТЬ

СКОРОСТНАЯ ДВУСТОРОННЯЯ ПЕЧАТЬ БЕЗ КОМПРОМИССОВ!

Подчас выбор МФУ для бизнеса схож с классическим спором ангела и беса, где жажда производительности спорит с необходимостью экономить. Но зачем выбирать что-то одно, когда с новым МФУ Panasonic вы можете получить все и сразу?

Многофункциональное МФУ Panasonic обеспечит вас скоростной двусторонней печатью и при этом позволит экономить на стоимости каждой страницы! С ним извечный спор будет решен в вашу пользу!



KX-MB2230RU

факс/телефон/принтер/сканер/копир/PC-факс
сетевой интерфейс
двусторонняя лазерная печать – 28 стр./мин.
односторонняя лазерная печать – 28 стр./мин.
автоподатчик на 50 листов
сканирование на FTP-сервер (PDF, JPEG, TIFF)
экологический режим
работа с различными типами бумаги



KX-MB2270RU

факс/телефон/принтер/сканер/копир/PC-факс
беспроводная сеть Wi-Fi* (IEEE 802.11 b/g/n)
сетевой интерфейс
двусторонняя лазерная печать – 28 стр./мин.
односторонняя лазерная печать – 28 стр./мин.
автоподатчик на 50 листов
сканирование на FTP-сервер (PDF, JPEG, TIFF)
экологический режим
работа с различными типами бумаги



KX-MB2510RU

принтер/сканер/копир/
сетевой интерфейс
двусторонняя лазерная печать – 28 стр./мин.
односторонняя лазерная печать – 30 стр./мин.
двусторонний автоподатчик на 50 листов
сканирование на FTP-сервер (PDF, JPEG, TIFF)
прямая печать/сканирование с/на USB-накопитель
экологический режим
работа с различными типами бумаги



KX-MB2540RU

факс/телефон/принтер/сканер/копир/PC-факс
сетевой интерфейс
двусторонняя лазерная печать – 28 стр./мин.
односторонняя лазерная печать – 30 стр./мин.
двусторонний автоподатчик на 50 листов
сканирование на FTP-сервер (PDF, JPEG, TIFF)
прямая печать/сканирование с/на USB-накопитель
функция интернет-факса
экологический режим
работа с различными типами бумаги

■ ПРИНТЕР ● СКАНЕР ▲ КОПИР ▣ ФАКС

* Требуется беспроводной роутер для Wi-Fi подключения. Wi-Fi* является товарным знаком Wi-Fi Alliance.

www.panasonic.com mfu.panasonic.ru

Информационный Центр Panasonic: для Москвы 8-495-725-05-65, для регионов РФ 8-800-200-21-00 (звонок бесплатный)
На правах рекламы ООО «Панасоник Рус» — уполномоченного представителя компании Panasonic Corporation Ltd. на территории России



KASPERSKY®

МАЛИНА ДЛЯ АДМИНА



Надежная защита — сладкая жизнь!

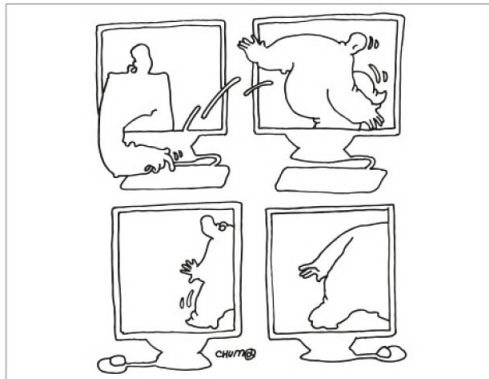
«Малина для админа» — специальная бонусная программа для системных администраторов. Сделайте бизнес ваших клиентов максимально безопасным с помощью надежных антивирусных решений «Лаборатории Касперского». Регистрируйте номера лицензий, накапливайте бонусные баллы и обменивайте их на отличные подарки. Чем больше компаний вы защитите, тем больше подарков получите!



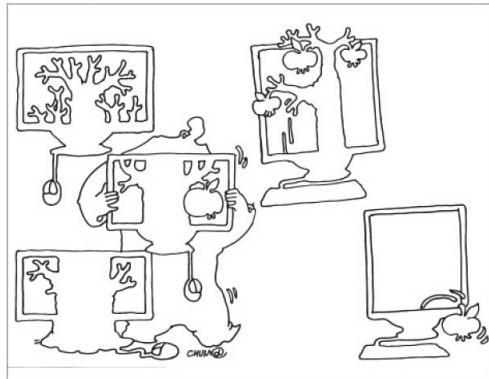
Подробнее: malina.kaspersky.ru

© ЗАО «Лаборатория Касперского». 2014. Зарегистрированные товарные знаки и знаки обслуживания являются собственностью их правообладателей.

Реклама



04



30

Острый угол

- 04** **Тотальный шпионаж и кибервойна, или Как АНБ хочет завоевать мир с помощью вирусов.** Человечество сегодня оказалось на грани новой войны – кибернетической.

Владимир Брюков

АДМИНИСТРИРОВАНИЕ

Документооборот

- 08** **Как выбрать СЭД?** Информации по СЭД много, но, когда приступаешь к выбору, оказывается, что ее недостаточно. Описания решений мало чем различаются, а отзывы на форумах противоречат друг другу.

Сергей Яремчук

Мониторинг

- 12** **Проверка на соответствие параметров, или Compliance Settings в SCCM 2012.** Compliance Settings в SCCM 2012 помогает оценивать и отслеживать достоверность конфигураций компьютеров компании, а также вносить в них некоторые исправления.

Сергей Болдин

FAQ

- 17** **Настройки сети с помощью PowerShell 4.0.**

Сергей Яремчук

Управление ИТ-инфраструктурой

- 18** **Не теряя управления. Часть 2. Обзор средств удаленного управления.** Создание отказоустойчивых решений, предотвращающих потерю управления сервером в критичной ситуации.

Алексей Бережной

How To

- 23** **Набор утилит dtools. Управляем множеством UNIX систем.**

Сергей Яремчук

Инструменты

- 24** **Инфокиоск своими руками.** Как подготовить «общественный компьютер» для работы в роли тонкого клиента и защитить его от нецелевого использования.

Игорь Орещенков

- 30** **Установка и настройка Katello, Pulp и Candlepin. Знакомимся с новым поколением Red Hat Network.**

Следующее поколение Red Hat Network – Katello, Pulp, Candlepin и Foreman. Познакомимся с первыми тремя приложениями.

Денис Силаков

- 35** **Системы управления конфигурацией. Чем отличаются Chef, SaltStack, Puppet, CFEngine и Ansible?** Четыре важнейшие группы свойств систем управления конфигурацией, по которым делается сравнение популярных CM-инструментов.

Сергей Житинский, Александр Чистяков

Виртуализация

- 40** **Облако от VMware. Часть 1. Разворачиваем vCloud Director.** Рассмотрим набор продуктов vCloud Suite, его базовые блоки и компоненты, а также установку vCloud Director с описанием всех шагов.

Александр Руденко

Копаем глубже

- 46** **USB_ModeSwitch, или «Кот в мешке».** Подключили USB-сотовый модем к ОС Linux, а он не заработал? Почему это случилось и что делать?

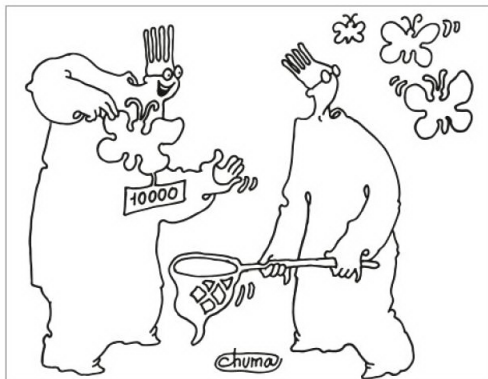
Павел Закляков

БЕЗОПАСНОСТЬ

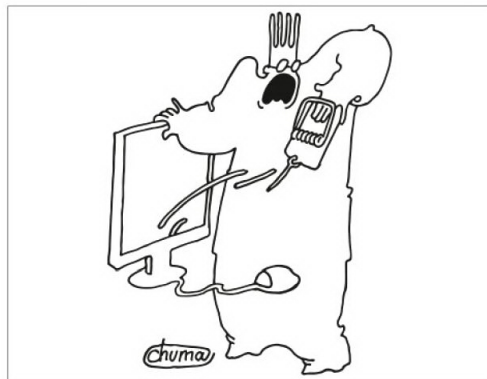
Мониторинг

- 52** **Большой Брат видит тебя.** Программы для скрытого наблюдения за пользователем, позволяющие перехватывать его почту и символы, вводимые с клавиатуры, контролирующие использование сменных носителей и делающие скриншоты его экрана, записывающие переговоры по скайпу.

Рашид Ачилов



80



84

БАЗЫ ДАННЫХ

Изучаем «1С»

- 59** Как избежать ошибок учета в «1С». Часть 11. Налог на добавленную стоимость. Учет налога на добавленную стоимость регламентируется законодательными документами. Поэтому важно обеспечивать корректность операций.

Игорь Чуфаров

- 66** «1С:ERP 2.0». Что нового? Часть 3. Подсистемы «Финансы» и «Бюджетирование». Продолжаем изучение нового флагманского решения от 1С.

Олег Филиппов

РАЗРАБОТКА

Инструменты

- 70** Программирование USB в Android. Используем для связи интерфейс USB. Организация обмена информацией между электронными устройствами и аппаратами, работающими под управлением ОС Android с использованием USB.

Сергей Ильичев

Проектирование

- 76** Очереди. Теория и практика. Ранее мы рассмотрели взаимодействие потоков с использованием очередей в рамках одного адресного пространства. А как организовать взаимодействие для нескольких процессов, даже если они разнесены на разные физические серверы?

Александр Календарев

Веб-технологии

- 78** Меньше кода, больше дохода. Часть 4. Разработка front-end-слоя. В предыдущих частях статьи были изучены вопросы, связанные с разработкой back-end-слоя, использованием базовых возможностей и т.д. Поговорим о применении JavaScript и CSS-библиотек (front-end-слой) в Apache Tapestry веб-приложении.

Михаил Ушаков

КАРЬЕРА/ОБРАЗОВАНИЕ

Пятая пара

- 80** Бесплезные занятия. Проблемы российского ИТ-образования. Взгляд изнутри. Предложения по его совершенствованию.

Дмитрий Андриенко

Образование в сети

- 82** Программирование на Java. Для тех, кто желает познакомиться или углубить свои знания, мы предлагаем перечень курсов.

Игорь Штомпель

Рынок труда

- 84** Безопасное трудоустройство. Часть 5. Где лучше работать российским ИТ-специалистам. Рекомендации для рассмотрения тех или иных «заманчивых» предложений о работе.

Владимир Иванов

- 88** Вакансия: тестировщик ПО. Мы спросили у представителей компаний, каким требованиям должен удовлетворять кандидат.

Игорь Штомпель

- 91** От Москвы до Бреста не прожить без тестов. Исследовательский центр рекрутингового портала Superjob.ru подготовил обзор предложений и требований работодателей, ищущих тестировщиков ПО.

Валерия Чернецова

Ретроспектива

- 92** Прямой путь к богатству. Напрямую без посредников. Это позволило Майклу Деллу вывести свою компанию в мировые лидеры.

Владимир Гаков

Зал славы «СА»

- 96** Майкл Делл. О бизнесе и месте в жизни.

Владимир Гаков



Визитка

ВЛАДИМИР БРЮКОВ, независимый аналитик

Тотальный шпионаж и кибервойна, или Как АНБ хочет завоевать мир с помощью вирусов

Человечество, пережившее две мировые войны, похоже, сегодня оказалось на грани начала третьей — кибернетической, хоть и не столь кровавопрлитной, но тоже весьма разрушительной

Атака на Иран

В условиях растущей автоматизации и интернетизации современного общества атака уже в совсем недалеком будущем может начаться не с выстрелов или залпов артиллерийских орудий, а с внезапной и незаметной атаки с помощью кибероружия, способного быстро и эффективно уничтожить цель. А затем... просто удалить себя, оставив неприятеля в неведении не только по поводу причины произошедшего ЧП, но и авторства страны-агрессора.

Причем ряд признаков надвигающегося глобального кибернетического противоборства уже сегодня нетрудно заметить. Во-первых, уже сформировался союз стран в виде разведывательного альянса «Пять глаз» с участием США, Великобритании, Канады, Австралии и Новой Зеландии, который последние пять лет не только активно занимается сбором информации, но и наносит киберудары по своим противникам. К этому сообществу следует добавить Израиль — эта страна хотя и в альянс и не входит, но с американцами тесно сотрудничает.

Во-вторых, появилась и первая страна-жертва, пострадавшая от разрушительных кибератак. Как и следовало ожидать, о первом акте киберагрессии мировая общественность узнала не от ее инициаторов, а из очень краткого сообщения на сайте WikiLeaks от 17 июля 2009 года: «Две недели назад источник, связанный с ядерной программой Ирана, конфиденциально сообщил WikiLeaks о серьезной аварии в Нетензе, являющейся главным иранским центром по обогащению урана. ...WikiLeaks обычно не сообщает о таком инциденте без дополнительного подтверждения, но, по данным иранских СМИ и британской радиовещательной корпорации BBC, глава иранской Организации по атомной энергии Голям Реза Агазаде ушел в отставку при весьма загадочных обстоятельствах... Согласно сообщениям СМИ, в отставку он подал еще 20 дней назад» [1].

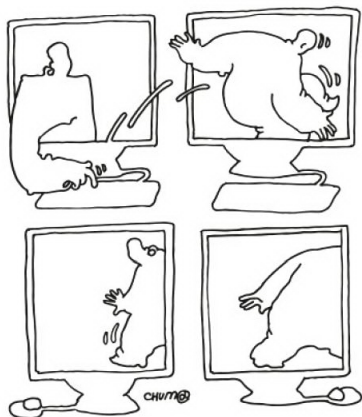
Подготовка к кибератаке на иранский центр по обогащению урана в Нетензе началась еще в 2006 году, когда выяснилось, что многие союзники США не готовы к согласованным санкциям против Ирана. Тогда заместитель председателя Объединенного комитета начальников штабов

генерал Джеймс Картрайт представил президенту Дж. Бушу-младшему компьютерную программу, которую можно было использовать в качестве наступательного кибероружия. При этом перед АНБ тогдашней президентской администрацией была поставлена цель получить доступ к промышленным контроллерам завода в Нетензе, но для этого требовался программный код, способный инфицировать компьютеры этого предприятия, не имеющие связи с Интернетом. Для нанесения эффективного киберудара код нужно было ввести в специализированные компьютеры, управлявшие центрифугами. Операция проводилась совместно с радиоэлектронной разведкой Израиля, собравшей довольно подробную информацию о работе центра в Нетензе [2].

С первых месяцев пребывания в своей должности президент Обама тайно приказал усилить кибератаки на компьютерные системы, управляющие на иранских заводах процессом обогащения урана, тем самым впервые значительно расширив использование в США этого вида оружия. На это решение не повлиял даже тот факт, что из-за программной ошибки вирус вышел за пределы иранского центра в Нетензе и начал распространяться по миру. В результате эксперты по ИТ-безопасности приступили к исследованию обнаруженного ими «боевого червя», созданного в США и Израиле, которому было дано имя Stuxnet [2].

По мнению эксперта натовского киберцентра в Эстонии Кеннета Гирса, высказанному на одной из конференций о безопасности, успех атаки Stuxnet зависел исключительно от контактов с нужными людьми и... от элементарных USB-накопителей. «Можно заплатить кому-то, кто запустит трояна в закрытую систему или подменит флешку, которая предназначалась только для внутреннего пользования. Достаточно вставить в стандартный USB-разъем компьютера инфицированную флешку, и Stuxnet тут же автоматически перескакивает на операционную систему, и никакие антивирусные программы и прочие меры защиты ей не помеха», — полагает г-н Гирс [3].

И действительно, «слабым звеном» оказался человеческий фактор — Stuxnet был занесен в систему посредством обычного USB-накопителя, который по неосторожности



В условиях глобализации кибершпионажа в зону риска попадают даже простые интернет-пользователи

вставил в рабочую станцию нерадивый сотрудник. Примечательно, что после заявления министра разведки Ирана Гейдара Мослехи о задержании «ядерных шпионов» (ими оказались совершенно непричастные к этому российские техники) руководство Siemens признало, что вирус занесли сотрудники компании, подчеркнув непреднамеренный характер заражения. Следует отметить, что Stuxnet поражает лишь конкретный тип контроллеров Siemens, а именно SIMATIC S7, который, по сведениям МАГАТЭ, используется лишь Ираном [3].

Судя по данным СМИ, Stuxnet примечателен тем, что использовал не одну, а четыре 0-day-уязвимости операционных систем семейства Windows, из которых только две были известны экспертам по ИТ-безопасности, но очень мало. Кроме того, вирус использовал еще и пятую, известную, но очень серьезную уязвимость в службе удаленного вызова процедур Windows (RPC), которую уже использовал червь Conficker. Вирус был подписан краденой цифровой подписью. Вредоносный код размером в половину мегабайта писала команда настоящих профессионалов на ассемблере, С и С++. При этом вирус не занимается рассылкой спама, не форматирует диск и даже не крадет банковские данные, а атакует промышленные системы контроля и управления, использующие софт Simatic WinCC [4]. Целью кибернападения было заставить изменить скорость движения ротора центрифуги, которая сначала поднималась до 1410 герц, потом снижалась до 2 герц и вновь поднималась до нормальной – 1064 герц [5]. В результате столь резких перепадов в скорости вращения центрифуги выходили из строя.

Впрочем, как полагают эксперты, если целью было быстро уничтожить все центрифуги, то ее Stuxnet не достиг. Но если целью было уничтожить ограниченное число центрифуг и замедлить прогресс Ирана в деле обогащения урана, то ее можно считать, хотя бы временно, достигнутой [6].

Статистические данные, опубликованные Федерацией американских ученых (the Federation of American Scientists) показывают, что число работающих центрифуг в Нетензе сократилось с 4700 до 3900 единиц в первой половине 2009 года, и, по всей видимости, вирус мог уничтожить

до 1000 центрифуг в период между ноябрем 2009-го и концом января 2010-го [7].

Бывший сотрудник Агентства национальной безопасности (АНБ) Эдвард Сноуден в интервью немецкому еженедельнику «Шпигель» 9 июля прошлого года подтвердил факт сотрудничества США и Израиля в разработке «боевого червя» Stuxnet, заметив, что он был создан для атаки ядерной программы Ирана [8].

Слежка во вселенском масштабе

Обнародованные Сноуденом документы также свидетельствуют о том, что АНБ планирует заразить миллионы компьютеров по всему миру вредоносным софтом. Об этом недавно написали американские авторы Райан Галлахер (Ryan Gallagher) и Гленн Гринвальд (Glenn Greenwald), которым беглый сотрудник АНБ передал всю позаимствованную им у агентства секретную документацию [9].

Как пишут эти авторы, АНБ начало быстро расширять свои хакерские возможности еще лет десять назад. Согласно внутренней секретной информации агентство еще в 2004 году создало небольшую хакерскую сеть, способную взломать и вести слежку в масштабе 100-150 компьютеров. Но в течение следующих шести-восьми лет ТАО (Tailored Access Operations) – элитное подразделение АНБ, занимающееся операциями специального доступа, – сумело набрать новых хакеров и разработать новые инструменты для взлома. В результате этого количество атакуемых вирусом шпионами компьютеров увеличилось до десятка тысяч.

Для того чтобы проникнуть в зарубежные компьютерные сети, на которые не было доступа с помощью других средств, АНБ решило выйти за пределы традиционной радиоэлектронной слежки (SIGINT), заключающейся в перехвате сигналов, посылаемых по каналам связи. Вместо этого оно стремилось расширить «активные» методы слежки, используя тактику прямого внедрения вредоносного ПО в компьютеры и сетевые устройства.

В опубликованных Сноуденом документах агентство описывает такие методы, как «более агрессивный подход к радиоэлектронной слежке SIGINT», и говорит о том, что задачей

его спецподразделения ТАО является резкое расширение масштабов таких операций. Вместе с тем АНБ признает, что управление масштабной сетью внедренных вирусов-шпионов является слишком сложной задачей для человека. «Одной из самых серьезных проблем для активных SIGINT-атак является их масштаб. Человек не в состоянии управлять масштабной сетью внедренных вирусов-шпионов, поскольку люди, как правило, действуют в рамках своей собственной среды, не принимая во внимание общую ситуацию», – говорится в сверхсекретном документе АНБ 2009 года [9].

Эту проблему агентство решило с помощью системы TURBINE, реализованной в рамках деятельности его спецподразделения ТАО. Как это сообщается в документах, обнародованных Сноуденом, TURBINE представляет собой совокупность интеллектуальных средств контроля и управления за крупномасштабной эксплуатацией сетью внедренных вирусов-шпионов. Таким образом, система TURBINE была создана с целью облегчить использование вредоносного ПО для хакеров АНБ.

Как сказано в одном из документов, подготовленных дирекцией АНБ, данная система позволяет избавить ее пользователя от необходимости знать или заботиться о деталях, связанных с проводимой им операцией: «Например, пользователь может запросить все детали о приложении X, но ему не нужно знать, как и где это приложение хранит файлы, записи в реестре и прочие данные» [9].

На практике это означает, что TURBINE автоматизировала многие важные процессы, которые ранее приходилось выполнять вручную, в том числе заниматься конфигурацией внедряемых шпионов-вирусов и постановкой задач, связанных со сбором разведывательных данных из зараженных систем. Причем сама по себе процедура автоматизации этих процессов имела не только технический характер, но и была чем-то большим. Этот шаг представлял собой серьезный сдвиг в деятельности АНБ, позволяющий ей значительно расширить границы электронной слежки. О его последствиях подробно рассказывается в одном недатированном сверхсекретном документе АНБ, в котором описано, как за счет системы TURBINE агентство планирует увеличить количество шпионского софта с нескольких сотен до потенциально возможных миллионов копий. Причем в шпионском ПО имелись как программы CNE (Computer Network Exploitation), внедряемые в компьютеры и сети в целях сбора информации, так и программы CNA (Computer Network Attack), предназначенные для атаки, нанесения ущерба либо разрушения их.

Обнародованные секретные данные позволяют сделать вывод, что внедрение системы TURBINE дало свои результаты. Данная система стала функционировать примерно с июля 2010 года, постепенно занимая все более важное место в хакерских операциях АНБ. Согласно обнародованным Сноуденом данным, АНБ уже удалось внедрить примерно 85 000–100 000 копий вредоносного софта в компьютерах и сетях по всему миру.

Следует заметить, что система TURBINE работает с ведома и поддержки ряда других стран, часть которых принимала участие в распространении вредоносных программ. Специальная маркировка на рассекреченных Эдвардом Сноуденом документах показывает, что они предназначались не только для США, но и для Великобритании, Канады, Новой Зеландии и Австралии, которые входят в альянс «Пять глаз» [9].

Изохронные способы обхода защиты

АНБ имеет разнообразный арсенал вредоносных инструментов, высокоинтеллектуальных и настраиваемых для различных целей. Одна шпионская программа под кодовым названием UNITEDRAKE может быть использована с различными «плагинами», позволяющими агентству получить полный контроль над зараженным компьютером.

Например, внедряемый плагин по имени CAPTIVATED-AUDIENCE используется, чтобы взять под контроль микрофон компьютера и записывать разговоры, происходящие рядом с устройством. Другой плагин GUMFISH может секретно взять под контроль веб-камеру компьютера и делать фотографии. FOGGYBOTTOM записывает журнал посещений используемого интернет-браузера, а также собирает информацию о его логинах, паролях, доступе на веб-сайты и учетные записи электронной почты. Плагин GROK используется для перехвата вводимых с клавиатуры символов. А плагин SALVAGERABBIT извлекает данные из USB-флеш-накопителей, подключаемых к пораженному компьютеру [9].

Шпионские вирусы позволяют АНБ обходить используемые в Интернете инструменты защиты, необходимые для сохранения анонимности пользователей, а также вскрывать рассылаемые по всей сети зашифрованные письма. Внедренный шпионский софт дает агентству неограниченный доступ к взломанному компьютеру, и информация о содержании письма становится известна еще до его шифровки. Неясно, сколько вредоносного ПО ежегодно внедряется в компьютерные сети, и какие модификации вирусов-шпионов сегодня наиболее активно используются АНБ по всему миру.

У АНБ были также вирусы, специально предназначенные для заражения крупных сетевых маршрутизаторов, используемых интернет-провайдерами за рубежом. Благодаря этому агентство получало секретный доступ и могло отслеживать интернет-трафик и журнал посещений пользователя, а также перехватывать его сообщения.

Как показывают обнародованные секретные данные, далеко не все вирусы использовались для сбора информации. Иногда АНБ ставило перед собой цель не организации слежки, а реализации атаки на компьютерную сеть пользователя. Например, программа QUANTUMSKY, разработанная еще в 2004 году, предназначалась для блокировки атакуемого компьютера от доступа на определенные сайты. Другая программа QUANTUMCOPPER, впервые протестированная в 2008-м, блокирует загрузки файлов у атакованного компьютера. Согласно обнародованным Сноуденом данным, из девяти вредоносных программ, разработанных АНБ, шесть были предназначены для слежки, две – для атаки и только одна использовалась в целях защиты сети правительства США.

Согласно секретному документу от 2012 года АНБ могло распространять свое вредоносное ПО путем рассылки спама. После того как пользователь открывал это письмо, его компьютерная сеть заражалась в течение восьми секунд.

Правда, поскольку интернет-пользователи стали опасаться открывать подозрительные письма от неизвестных им лиц, АНБ стала применять новые технологии взлома. В частности, одну из таких технологий называют «атака посредника». Ее суть заключается в том, что секретно подклю-

чившийся к каналу связи криптоанализа способен читать и видоизменять сообщения, которыми обмениваются корреспонденты.

В свою очередь, еще одна новая технология взлома называется «атака человека со стороны». С этой целью АНБ следит за интернет-трафиком пользователя до тех пор, пока он не входит на сайт, который доступен ее атаке. В этот момент установленные на зараженном сайте сенсоры дают сигнал системе TURBINE, которая в долю секунды нацеливается на IP-адрес компьютера жертвы. В результате чего происходил скрытый взлом компьютера, с жесткого диска которого извлекалась вся имевшаяся там информация. Для более эффективного использования «атаки человека со стороны» АНБ даже пришлось создать ложный сервер Facebook.

Чтобы внедрить некоторые из своих шпионских программ, АНБ использует уязвимости в системе безопасности в таких популярных интернет-браузерах, как Mozilla Firefox и Internet Explorer. Хакеры агентства также используют слабые места в безопасности сетевых маршрутизаторов и в популярных плагинов, таких как Flash и Java, для доставки вредоносного кода в предназначенные для взлома компьютеры.

Внедряемые вирусы в состоянии обойти антивирусные программы, причем АНБ приняла меры по сохранению секретности используемой технологии. Например, внедряемая вирусная программа VALIDATOR, которая использует внедрения вирусов и загрузки данных с зараженного компьютера, может самоуничтожиться после истечения установленного периода времени.

Во многих случаях брандмауэры и другие средства защиты, по всей видимости, не представляют большого препятствия для АНБ. «Если пользователь намеченного к взлому компьютера вынужден будет посетить нас (то есть зараженный сайт – прим. Б.В.) с помощью определенного типа веб-браузера, то мы с ним скорее всего справимся, единственный только вопрос: как?» – хвастается один из хакеров в опубликованном секретном документе.

Шпионаж за сисадминами

По информации Сноудена, описанные выше технологии слежки использовались как против подозреваемых в терроризме, так и против лиц, рассматриваемых АНБ в качестве экстремистов. Однако мандат на взлом, выданный хакерам агентства, не ограничивался только теми, кто представлял угрозу безопасности США.

В одном секретном служебном сообщении для внутреннего пользования один из руководителей АНБ описывает использование вредоносного ПО для атаки на компьютеры системных администраторов, работающих в зарубежных телефонных компаниях и у провайдеров интернет-услуг. Путем взлома компьютера сисадмина агентство может получить скрытый доступ к обширной информации о клиентах, имеющейся у этой компании. Таким образом, охота на компьютеры сисадминов облегчала для АНБ организацию электронной слежки, например, с правительственными чиновниками [9].

Подобная тактика использовалась и руководством Government Communications Headquarters (GCHQ), британской спецслужбой, выполняющей такие же задачи, как и АНБ. По информации немецкой газеты «Шпигель»,

опубликованной в сентябре прошлого года, сотрудники GCHQ взломали компьютеры, принадлежащие сетевым инженерам Belgacom – бельгийской телекоммуникационной компании. Взлом под кодовым названием «Операция социалист» имел своей целью мониторинг звонков, поступающих с мобильных телефонов, подключенных к сети Belgacom. Опубликованные секретные данные говорят о том, что благодаря этой операции британская спецслужба имела доступ к сети Belgacom по меньшей мере с 2010 года.

АНБ ищет новые жертвы, за которыми устанавливается наблюдение, с помощью специальных фильтров, отслеживающих информацию, циркулирующую в интернет-кабелях. При этом фильтрация информации согласно опубликованным секретным документам производится с учетом таких данных, как адреса электронной почты, IP-адреса, и уникальных куки (небольших фрагментов данных, отправленных веб-сервером и хранимых на компьютере пользователя), содержащих имя пользователя или другую идентифицирующую информацию, которая посылается пользователю такими сайтами, как Google, Facebook, Hotmail, Yahoo и Twitter.

Другую информацию АНБ может получить из рекламных куки Google, отслеживающих привычки, уникальные ключи шифрования и идентификаторы компьютера, отправляемые по Интернету во время перезагрузки компьютера или обновления операционной системы Windows.

Следовательно, сегодня практически любой пользователь Интернета, живущий в том или ином конце мира, с той или иной вероятностью рискует стать объектом слежки для АНБ. Причем защититься от назойливого интереса этой разведслужбы довольно трудно, поскольку она обладает широким набором инструментов для обхода даже самых стойких средств защиты. Более того, в условиях глобализации кибершпионажа в зону риска попадают даже среднестатистические интернет-пользователи, до которых у АНБ раньше просто не доходили руки... **Бор**

1. Serious nuclear accident may lay behind Iranian nuke chief's mystery resignation на сайте http://wikileaks.org/wiki/Serious_nuclear_accident_may_lay_behind_Iranian_nuke_chief%27s_mystery_resignation.
2. The New York Times, Sanger, David. Obama Order Sped Up Wave of Cyberattacks Against Iran (1 June 2012).
3. Stuxnet: война 2.0 от 12 октября 2010 г. на сайте <http://habrahabr.ru/post/105964>.
4. Федеральная Служба Опасности. Про вирус Stuxnet от 18 сентября 2010 года на сайте <http://malaya-zemlya.livejournal.com/584125.html>.
5. Eric Chien. Stuxnet: A Breakthrough. Symantec. Retrieved 14 November 2010.
6. Did Stuxnet Take Out 1,000 Centrifuges at the Natanz Enrichment Plant? Institute for Science and International Security. 22 December 2010. Retrieved 27 December 2010.
7. IAEA Report on Iran. Institute for Science and International Security. 16 November 2010. Retrieved 1 January 2011.
8. Snowden confirms NSA created Stuxnet with Israeli aid на сайте <http://rt.com/news/snowden-nsa-interview-surveillance-831>.
9. How the NSA Plans to Infect «Millions» of Computers with Malware. By Ryan Gallagher and Glenn Greenwald 12 Mar 2014 на сайте <https://firstlook.org/theintercept/article/2014/03/12/nsa-plans-infect-millions-computers-malware>.



Визитка

СЕРГЕЙ ЯРЕМЧУК, автор более 800 статей и шести книг.
С «СА» с первого номера. Интересы: сетевые технологии,
защита информации, свободные ОС, grinder@samag.ru

Как выбрать СЭД?

Информации по СЭД вроде бы и много, но, когда приступаешь к выбору, оказывается, что ее недостаточно. Описания решений мало чем различаются, а отзывы на форумах противоречат друг другу

Сегодня решение о переходе на систему документооборота редко принимается спонтанно, без каких-либо конкретных планов или как дань моде, просто «чтобы было». Когда количество бумаг начинает превышать определенные пределы, а найти что-то становится сложно, начальство само приходит к мысли о необходимости перемен. Проект внедрения СЭД начинается с постановки задачи, определения его рамок и выбора программной платформы.

Некоторые вендоры не сильно акцентируют внимание на функциях продукта, более важную роль играют бренд и наличие известных клиентов. Ведь ИТ-системы, и СЭД в том числе, считаются сложными, и разобраться в них может только специалист. Поэтому вариант, что продукт будут изучать перед внедрением, учитывают не все разработчики и стараются действовать старыми методами, продавая скорее свою репутацию/бренд, а не показывая функционал.

Посетителя на сайтах встречает маркетинговая информация, а при поисках конкретики складывается мнение, что некоторые разработчики как будто бы специально что-то скрывают. Хотя это может быть и свидетельством того, что решение в последующем будут адаптировать под конкретную задачу. Определиться в такой ситуации очень сложно, часто нужно перебрать груды документации, чтобы докопаться до сути.

Нет, конечно, менеджеры по продаже ответят на любой ваш вопрос, но, правда, в пределах своей компетенции, «мелочей» они не знают. Ведь не секрет, что разрабатывают систему одни, продают и внедряют другие, а поддерживают третьи. У каждого круга свой уровень знаний, и добраться до самой сути тяжело (хотя в этом не всегда есть необходимость).

Для ускорения процесс внедрения полностью отдается стороннему поставщику, который произведет аудит и предложит решение. Вот этот этап вызывает вполне обоснованные сомнения, так как, как правило, ИТ-консалтинг и внедрение проводит одна и та же компания, то есть решение уже «выбрано» заранее, просто его немного адаптируют к внутренним нуждам. Доверившись сторонней компании, легко попасться на маркетинговый крючок, поведаясь на вну-

шительный список внедрений и отзывов благодарных клиентов. Конечный функционал представят уже после внедрения.

Современный руководитель вполне способен самостоятельно оценить предлагаемый продукт, прежде чем сделать выводы о целесообразности его внедрения. Нет, конечно, он вряд ли будет смотреть на такие характеристики, как системные требования, используемая СУБД или язык, на котором написана программа. Речь именно о функциональности СЭД, с которой придется работать обычному сотруднику и на детали которой редко обращает внимание айтишник, так как он имеет только общее представление о бизнес-процессах. А вот этот момент часто не учитывают.

Итак, чтобы выбрать СЭД, нам нужно познакомиться с общей функциональностью и разобраться, как реализованы отдельные процессы (например, согласование документов) и возможности по изменению маршрутов.

В этом нам должны помочь общее описание продукта (введение) и документация. Здесь нам интересно увидеть максимальные детали в описании именно функционала СЭД, то есть что мы можем сделать или получить.

Присутствие общих маркетинговых фраз или акцент на ненужные мелочи технического характера, наличие большого количества подпунктов говорит о том, что вендор или пытается выдать продукт за другой, или банально не может написать нормальную документацию.

Лишними будут многочисленные статьи на сайте о необходимости использования СЭД. В них нет особого смысла, так как человек уже в принципе принял решение и приходит выбирать.

Очень кстати, если это все дополняется подробными скриншотами и демонстрационным видео, показывающим основные этапы работы программы. Ведь написать можно что угодно, но лучше просто увидеть, а снимок четко покажет, что и как, и поможет составить мнение. Чем больше такого материала, тем, наверное, лучше.

Бывает, что разработчики выкладывают маркетинговое видео, где счастливые пользователи рассказывают, как хорошо, что у них внедрена именно эта система, не объясняя,

по сути, почему. Подобное занимает время и отбивает охоту изучать продукт. Более подробно тот или иной процесс должен быть расписан в документации.

И весьма неплохо, чтобы была и демоверсия продукта (желательно SaaS, так как не всегда есть возможность настроить и показать всем заинтересованным), к которой можно было бы получить доступ без всяких бюрократических проволочек. Некоторые разработчики предоставляют свои решения для тестов после определенных согласований только через несколько дней, и, чтобы просто оценить, приходится разворачивать всю инфраструктуру (сервер, СУБД и т.п.).

Кстати, сама по себе SaaS-версия – очень неплохое начало для небольших компаний, которые не хотят или не могут развернуть полноценный сервис у себя на площадке. Кроме того, часто можно услышать жалобы на то, что решение «тормозит», но с SaaS при наличии каналов достаточной пропускной способности такое не наблюдается, так как разработчики знают, сколько нужно выделить ресурсов.

Ну и, конечно, мы должны получить внятное описание формирования цены на продукт. Для небольших организаций интересными будут решения SaaS, пользователю нет особой разницы, где именно находятся серверы и как именно хранятся документы, а вот процесс администрирования заметно упрощается.

Попробуем по этим критериям оценить доступную информацию по трем «чистым» СЭД-решениям. Выбрались они для обзора по наличию бесплатной версии.

Alfresco

Об Alfresco [1] журнал уже писал [2]. Эта СЭД разрабатывается с 2005 года и обладает всем необходимым для управления документами, записями, веб-публикацией и бизнес-процессами в организации. В магическом квадрате Gartner, посвященном решениям по управлению корпоративным контентом (ECM, Enterprise Content Management), Alfresco долгое время находится в Visionaries (провидцы). Продукт достаточно известен и получил несколько наград от разных организаций.

Сайт Alfresco англоязычный, поэтому базовые знания весьма желательны. На сегодня Alfresco представлен в двух версиях – Alfresco Community Edition и Alfresco One (ранее Enterprise), есть и SaaS-сервис Alfresco in the cloud, использующий веб-сервисы Amazon.

Собственно говоря, Alfresco стал известен именно благодаря наличию бесплатной версии, так как установить и использовать его может каждый желающий, благо в сети много руководств. Нередко именно эту версию предлагают некоторые компании-интеграторы.

Таблица показывает, что CE и One по функциям документооборота не отличаются, только по возможности расширения. В CE отсутствуют интеграция с облаком, кластеризация, поддержка некоторых бизнес-приложений (Salesforce), функция управления мобильными устройствами (MDM) и поддержка корпоративных платформ (MS SQL Server, Oracle и WebSphere). Сборка Community распространяется абсолютно свободно по лицензии GNU GPL без какой-либо поддержки. Enterprise ориентирована на корпоративный сектор и приобретает вместе с официальной поддержкой от разработчиков.

Выбрав нужный продукт, получим краткую информацию по возможностям. Например, для Alfresco One здесь семь пунктов, в которых раскрываются: управление документами, совместная работа, поддержка мультимедиа, управление бизнес-процессами, синхронизация контента, управление записями и инцидентами, поддержка открытых стандартов.

В принципе все важное здесь озвучено, этого достаточно, чтобы немного определиться. Остальные вопросы отражены в документации, где каждая версия продукта имеет отдельную ветку. Это плюс, так как нередко разработчики описывают только коммерческий продукт, а в том, чего не хватает в бесплатной версии, приходится разбираться самому.

Здесь находим технический обзор, архитектуру, информацию об API (RESTful API, Java API, JavaScript API), с помощью которых легко получить доступ к репозиторию, задачам, системе управления сайтами, данные по установке и использованию и многое другое.

Отдельный раздел представляет видеоруководства (всего 15), которые наглядно показывают процесс и подробно информируют о возможностях. В частности, их анализ, кроме базовых элементов, позволяет получить некоторое представление о настройке бизнес-процессов в Alfresco, модели контента, работе системы групповой работы и т.д. То есть как раз то, что объяснить на пальцах никак нельзя.

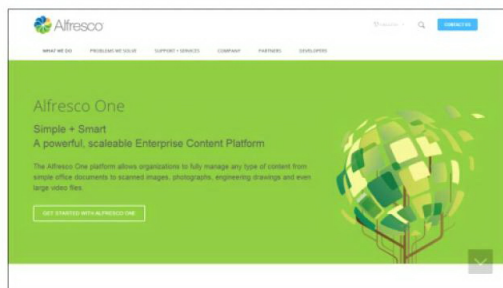
На сайте можно найти ссылки на White Papers, части опубликованных книг (требуется регистрация), блог разработчиков, Wiki и форум. Проект постоянно проводит вебинары, архив которых в свободном доступе.

Также на видном месте показана ссылка на дополнительные расширения Alfresco [3], которых в настоящий момент более 600. Здесь интеграция с соцсетями, YouTube, популярными CMS, языковые пакеты и многое другое.

Решение кроссплатформенное, разработано на основе Java-технологий, в качестве СУБД для свободной редакции поддерживаются MySQL и PostgreSQL. В редакции Alfresco One, также Oracle Database, DB2, MSSQL Server. Развернуть его при наличии определенного опыта вполне реально.

Также разработчики предлагают возможность сразу подключить Alfresco Cloud, для чего достаточно указать свой email. Вот здесь ложка дегтя. Запросы, очевидно, обрабатываются вручную, а поэтому некоторое время придется подождать (а можно и не дожидаться). Чтобы получить хоть какую-то информацию о стоимости One или Cloud, необходимо

Рисунок 1. Официальная документация Alfresco очень подробная, но, к сожалению, она на английском языке



заполнить лист с информацией о компании и подождать ответа менеджера.

Verdox

Сайт разработчика Verdox [4] сделан профессионально, но вот информация, которая первой попадает в поле зрения, на нем представлена типично маркетинговая.

Просмотрев первую попавшуюся в глаза информацию, мы узнаем, что с помощью этой системы можно «навести порядок в бизнесе». Не сразу удастся найти назначение продукта, о том, что это именно СЭД, узнаем, перемотав страницу до конца.

Но главный минус – на сайте нет никаких данных о компании-разработчике, просто в контактах номер телефона и адрес электронной почты. И все! Человеку, впервые попавшему на сайт, без каких-либо рекомендаций будет сложно доверять такой вот визитке. То есть раздел «О компании» сайту бы точно не помешал.

Ниже наконец находим что-то полезное. Здесь восемь пунктов, выбрав любой из них, можно получить подробную информацию, в которой расписано по пунктам, зачем все это нужно и чем поможет Verdox. Причем после введения приводятся конкретные данные с полноразмерными снимками.

Кроме этого, представлен Виртуальный тур. Это то, что действительно очень хорошо реализовано. Выбираем один из 17 пунктов и смотрим, как пошагово выполняется работа средствами Verdox. Мы с вами увидим процесс работы с документами: регистрацию, работу с электронным архивом, резолюции, поручения, согласования, протоколы совещания и др. Заодно складывается впечатление и об интерфейсе.

В наличии раздел документации (больше ориентированной на пользователя), Wiki и форум. Правда, судя по количеству тем и сообщений, он особо не пользуется популярностью, да и форуму, видимо, не более года. Компания имеет свой канал на YouTube, на котором представлены два видеозаписи: вводный и ознакомительный по работе с поручениями.

Из особенностей Verdox. Входящие сообщения связываются, даже если они были доставлены разными способами (электронным и бумажным), позволяя отслеживать переписку. Еще один плюс – исходящие письма могут быть от-

правлены по электронной почте прямо из системы Verdox, без использования внешнего почтового клиента.

В руководстве пользователя в общем показаны все стандартные для таких решений действия. Документация администратора позволяет узнать, как настроить роли, процесс согласования и справочники, бизнес-процессы, шаблоны и прочее. Ее следует изучить более тщательно.

Система предлагается в трех вариантах: Verdox Free (бесплатно, до пяти пользователей), Verdox Box (от 1000 руб.) и Verdox Cloud (от 250 руб.). Последние два можно попробовать.

Также показана внятная таблица цен в зависимости от количества пользователей и таблица сравнений возможностей версий. Например, Free, кроме отсутствия поддержки и ограничения по количеству пользователей, не содержит нескольких модулей.

Для доступа к демоверсии необходимо заполнить форму. Ответ приходит сразу же (только может попасть в спам). После этого около 15 минут формируется окружение, и будет выдана ссылка для входа, а на почту придут данные для регистрации. Теперь можно получить полное представление о продукте.

Самостоятельное развертывание версий Free и Box каких-либо проблем вызвать не должно. Для установки потребуется ПК под управлением Windows (в документации почему-то об этом не сказано, также нет и рекомендаций по версиям).

FossDoc

Разработка компании «ФОСС-Он-Лайн» [5], работающей на рынке ИТ-технологий с 1999 года, в его активах клиенты нескольких российских и украинских банков и госучреждений, статус Microsoft Gold Certified Partner.

Попав на главную страницу, сразу видим полезную информацию по назначению и основным плюсам этой системы: проста в развертывании, принцип лицензирования, гибкая настройка маршрутов, поддержка специфических функций для государственного и банковского секторов. Ничего лишнего, все по сути.

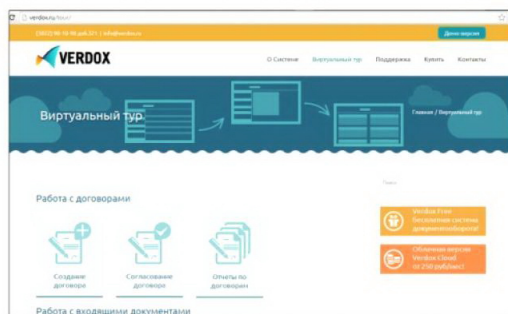
Буквально за пару минут, просмотрев основные ссылки с заглавной, можно получить первое впечатление о процессе работы с FossDoc.

Далее находим подробности, ссылку для выбора продуктов, расчет лицензии (возможна покупка и аренда) и ссылку на бесплатную версию. Этот момент особо отмечу, так как обычно информацию о наличии Free-версии пытаются спрятать подальше.

Бесплатная версия ограничена пятью учетными записями, чего обычно хватит для небольших организаций или для тех, кто только внедряет СЭД. Скачать ее можно свободно и без всяких регистраций, официальная поддержка в этом случае – только форум. Во многих бесплатных версиях коммерческого ПО установку обновлений стараются усложнить, в FossDoc они ставятся штатным образом (обновления выходят раз в месяц).

И еще есть интересная особенность. Подав запрос, можно бесплатно на 180 дней увеличить количество пользователей системы до 50. То есть можно спокойно внедрить систему, опробовать и затем уже купить лицензию. Версии SaaS в данный момент нет, но по сообщениям она находится

Рисунок 2. Виртуальные туры, предложенные Verdox, позволяют быстро понять процесс



в разработке. Для подключения к серверу потребуется установить клиентское приложение (только Windows).

Присутствует достаточная документация (для пользователя, администратора и разработчика), презентации, FAQ, форум поддержки и прочее. Имеется и свой канал на YouTube, на котором представлено 11 небольших роликов, наглядно демонстрирующих выполнение определенных действий с FossDoc.

Основной FossDoc является платформа FossLook [6], представляющая собой конструктор, позволяющий создавать собственные бизнес-приложения под любые задачи. СЭД FossDoc – уже готовое решение, включающее все необходимые модули и шаблоны документов для автоматизации классического делопроизводства.

Маршрут документа состоит из поручений исполнителям, которые приходят на рабочие места сотрудников. Открыв поручение, пользователь получает сам документ. Платформа интуитивно понятна, поэтому при необходимости стандартное решение легко адаптируется под любые условия (самостоятельно это сделать несложно): добавить любое поле, создать шаблон, прописать маршруты, публиковать документ в папку (архив).

Решение легко спрофилировать под любые другие задачи, организовать на основе FossDoc, например, складской учет. Реализован поиск по любым полям (расширенный и простой), фильтры папок, есть протоколы, в которых можно отследить историю правок любого документа.

Еще одна интересная функция. На основе FossDoc можно строить собственную корпоративную почту и держать полноценный почтовый домен, он может работать за релеем и забирать письма с других почтовых серверов (в том числе и бесплатных, вроде Mail.ru или Gmail). Принятое письмо автоматически превращается во входящий документ. Интегрирован с ЭЦП разных поставщиков.

Также документооборот интегрирован с видеоконференцией с помощью модуля от OpenMeetings [7]. Есть модули напоминаний, служебных записок и обращений граждан, позволяющие регистрировать и прописывать маршруты всем поступающим заявкам.

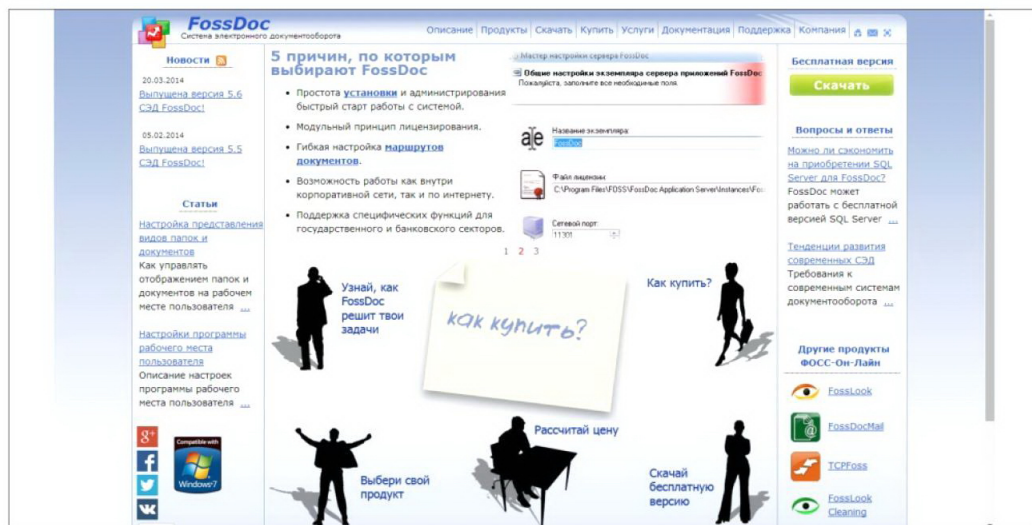
FossDoc рассчитан на использование в организациях, от небольших до крупных холдингов. Поддерживается интеграция с Active Directory, несколько СУБД (MS SQL, MySQL и Oracle Database). Установить можно на любую Windows, включая и клиентские XP SP3. Весь процесс расписан в документации, справиться сможет и подготовленный пользователь.

Итак, изучив документацию и протестировав продукты, мы уже можем самостоятельно определиться с требованиями к СЭД и выбрать те решения, которые больше под них подходят. **БОБ**

1. Сайт Alfresco – <http://www.alfresco.com>.
2. Яремчук С. Обзор Open Source ECM-системы Alfresco. // «Системный администратор», №3, 2009 г. – С. 37-43 (<http://samag.ru/archive/article/833>).
3. Дополнительные расширения Alfresco – <http://addons.alfresco.com>.
4. Сайт Verdox – <http://verdox.ru>.
5. Сайт FossDoc – <http://fossdoc.ru>.
6. Платформа FossLook – <http://fosslook.ru>.
7. Яремчук С. Система видеоконференций OpenMeetings. // «Системный администратор», №6, 2009 г. – С. 58-64 (<http://samag.ru/archive/article/2034>).

Ключевые слова: СЭД, документооборот, обзор, бесплатные решения.

Рисунок 3. На главной странице FossDoc сразу находим нужную информацию





Визитка

СЕРГЕЙ БОЛДИН, работает сисадмином в крупной энергетической компании НЭК Укрэнерго. Специализируется по MS SCCM, виртуализации, мониторингу, bsergey2@gmail.com

Проверка на соответствие параметров, или Compliance Settings в SCCM 2012

Compliance Settings в SCCM 2012 помогает оценивать и отслеживать достоверность конфигураций компьютеров компании, а также вносить в них некоторые исправления. В статье рассматриваются возможности и настройки данного функционала в теории и на практике

Часто случаются ситуации, когда необходимо получить точную информацию о состоянии рабочей станции или сервера, определить возникшую причину некорректной работы программы, которая могла быть подвержена обновлению или переустановке, а штатными средствами уследить за происходящими изменениями в системе не предоставляется возможным. Поэтому системный администратор применяет различные утилиты, например, NeoSpy [1], Process Monitor [2] и другие. Но они не всегда могут помочь в решении вопроса. Их основная задача – это предоставление информации о добавлении или модификации каких-либо файлов, записей в реестре, учетных данных пользователей и прочее. Главным недостатком таких программ является то, что приходится увеличивать количество уже имеющегося софта, а также отсутствие возможности редактирования конфигураций ПК. К тому же не все продукты поддерживают работу мобильных устройств. А так как у нас уже есть установленный Configuration Manager 2012, то рас-

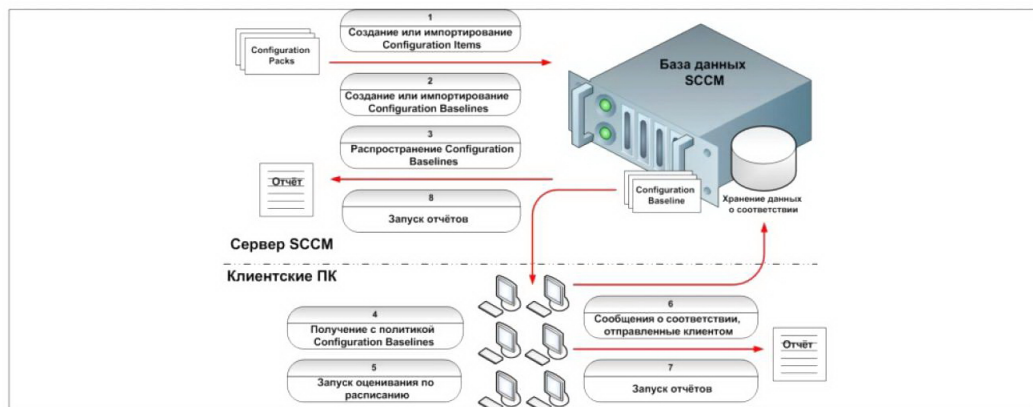
смотрим функцию Compliance Settings (параметры соответствия), которая позволяет не только устранить слабые стороны, перечисленные выше, но и предоставляет дополнительные преимущества.

Возможности

Compliance Settings [3] в SCCM 2012 предоставляет унифицированный интерфейс, который позволяет осуществить ревизию серверов, ноутбуков, настольных компьютеров и мобильных устройств на соответствие корпоративным конфигурациям. Данный функционал можно использовать для поддержки следующих бизнес-требований:

- > проверять конфигурации программ и пользовательских учетных записей, версии и обновления ОС и ПО, а также корректность установки и настройки сервера и сервисов;
- > автоматически исправлять несовместимые параметры WMI, реестра, сценариев;

Рисунок 1. Последовательность действий



- > выявлять устройства, которые не удовлетворяют корпоративным стандартам, то есть с неразрешенными процедурами управления изменениями, с наличием запрещенных приложений, а также с несовместимыми настройками перед вводом их в эксплуатацию или установкой на них ПО;
- > для отчетов устанавливать приоритеты несоответствия относительно пяти уровней серьезности: None (отсутствие), Information (информирование), Warning (предупреждение), Critical (критичность), Critical with event (критичность с событием);
- > идентифицировать уязвимости, определенные вендорами;
- > сравнивать конфигурации устройств предприятия с рекомендациями производителя;
- > вести совместную работу с другими продуктами управления, которые контролируют события Windows на компьютерах;
- > формировать отчет о соответствии нормам и корпоративным политикам безопасности.

Состав Compliance Settings

Compliance Settings включает в себя такие составляющие:

Configuration Item (Элемент конфигурации) – это единая конфигурация, которая содержит набор параметров, а также критерии проверки для оценки соответствия на целевой ОС.

Configuration Baseline (Базовый показатель конфигурации) – содержит элементы, которые необходимо оценить, а также параметры и правила, описывающие установленный уровень соответствия. Иными словами, это группа уже созданных Configuration Item и/или дополненная другими элементами, которая от сервера доставляется устройствам относительно установленного расписания. А если устройство не подключено к сети, но имеет загруженные Configuration Items, которые ссылаются на развернутые Configuration Baseline, то производится переподключение последних и оценка настроек на соответствие.

User Data and Profiles (Пользовательские данные и профили) – содержат параметры, определяющие, как управлять Folder redirection (Перенаправлением папок), Offline Files (Автономными файлами) и Roaming user profiles (Перемещаемыми профилями) на компьютерах под управлением Windows 8. А чтобы произвести развертывание на коллекции пользователей, нет необходимости добавлять их в Configuration Baseline, сделать распространение можно непосредственно с данного раздела.

Принцип работы Compliance Settings

Сначала на сервере SCCM происходит создание (или импортирование) Configuration Item, а затем их добавление в Configuration Baseline (или создание новых). Далее вся информация помещается в базу данных.

После этого на коллекцию пользователей или компьютеров распространяются Configuration Baseline. В свою очередь, клиенты SCCM получают от сервера политику с определенными параметрами или условиями и начинают проверку данной группы устройств на соответствие заявленным требованиям.

Запуск процедур выполняется по расписанию. А по завершении этого процесса отправляют информацию в БД. Заключительным действием является сформированный системным администратором отчет, который помогает судить о полученных результатах (см. рис. 1).

Возможные требования

Предположим, что необходимо проверить группу клиентских компьютеров на наличие архиватора 7zip и обновлений MS Excel 2013, а также сервер SCCM на соответствие последним обновлениям.

Но вначале проверяем, активирован ли параметр (агент) Compliance Settings в Administration → Overview → Client Settings → Default Client Settings, значение которого должно быть равным True.

Создание элемента конфигурации

Данный элемент создается следующим образом: заходим в Assets and Compliance → Overview → Compliance Settings → Configuration Item, в контекстном меню выбираем пункт Create Configuration Item. В появившемся мастере вводим требуемую информацию.

В разделе General (общие) это имя, тип элемента (мобильное устройство, ОС Windows, MAC OS) и категория (сервер, клиент, ИТ-инфраструктура или другая).

Если активировать флажок This configuration item contains application settings, то включим дополнительный раздел – Detection Methods (методы обнаружения), который содержит правила обнаружения приложения.

Наличие какой-либо программы определяется по присутствию у нее установочного msi-файла (узнаются код и версия продукта), а также использовать скрипт или всегда производить такое определение, выбрав первый вариант Always assume application is installed.

При проверке операционных систем в разделе Supported Platforms нужно определиться с версией ОС.

Рисунок 2. Результат создания Configuration Item

Icon	Name	Type	Revision	Relationships	Date Modified
	7zip	Operating System	2	Yes	4/23/2014 8:36 AM
	Microsoft System Center 2012 Configuration Manager Management Point	Application	2	Yes	4/23/2014 11:23 AM
	Microsoft System Center 2012 Configuration Manager Site Server	Application	1	Yes	4/23/2014 8:38 AM
	Microsoft System Center 2012 Configuration Manager Software Update P...	Application	1	Yes	4/23/2014 8:38 AM
	Windows Server Update Services configuration for Microsoft System Cent...	Application	1	Yes	4/23/2014 8:38 AM

В Settings выбираем тип проверки, а это могут быть значение или целиком ветка реестра, скрипт, файловая система, метабаза IIS, запросы SQL, Active Directory, WMI или другое. Compliance Rules предоставляет возможность создать и настроить условия для проверки соответствия элементов конфигурации относительно Value (значения) и Existential (существования).

В нашем случае создаем Configuration Item для архиватора, выбираем File System (файловая система), File (файл) и путь к файлу. Правило проверки создается по умолчанию, его можно отредактировать (если потребуется) или дополнить новым.

При необходимости создаем дочерний Configuration Item, где будут общие (родительские) параметры применяться на выбранную группу, и дополнительные – для специальных устройств, которые к общей массе не относятся. Для этого на панели задач нажимаем кнопку Create Child Configuration Item и добавляем готовый элемент с необходимыми условиями проверки.

Основное отличие по настройке соответствий для мобильных устройств располагается в разделе Mobile Device Setting groups. А количество вводимых параметров будет зависеть от того, сколько групп в нем было для этого выбрано. Здесь предлагается активировать следующее: Password, Email Management, Security, Peak Synchronization, Roaming, Encryption, Wireless Communication, Certificates, и внести нужные данные, создавая таким образом встроенные правила проверки соответствия, а дополнить их собственными условиями можно во вкладке Compliance Rules, зайдя в свойства элемента.

SCCM поддерживает несколько типов Configuration Items: Operating System (относится к ОС), Application (назначается программам), Software Updates (для обновлений), General (общие, не относящиеся ни к одному из предыдущих трех типов) (см. рис. 2).

После того как создан Configuration Item, над ним можно произвести некоторые действия. Например, экспортировать (кнопка «Export») и выгрузить в CAB-файл для дальнейшего переноса и импорта на другой сервер.

Также доступно: просмотреть исходный XML-файл (кнопка View XML Notification) или историю ревизий (кнопка History Revision), скопировать (кнопка Copy) или удалить (кнопка Delete) элемент.

Еще стоит обращать внимание на панель критериев, которая отображает полезную информацию (Type, Revision, Date Modified, Categories, Model Name, Status, Group by и многие другие) (см. рис. 2).

А для удобства предусмотрен поиск элементов по критериям нажатием кнопки Add Criteria и возможность сохранения этих результатов (кнопка Saved Searches).

Создание базового показателя конфигурации

Последовательность создания Configuration Baseline состоит из нескольких шагов: заходим в Assets and Compliance → Overview → Compliance Settings → Configuration Baselines, по нажатию правой кнопкой мыши требуется выбрать пункт Create Configuration Baselines. В появившемся мастере вводим необходимые данные, к которым относятся: имя, коллекция, единица конфигурации либо другие базовые показатели, либо обновления.

Рисунок 3. Результат создания Configuration Baseline

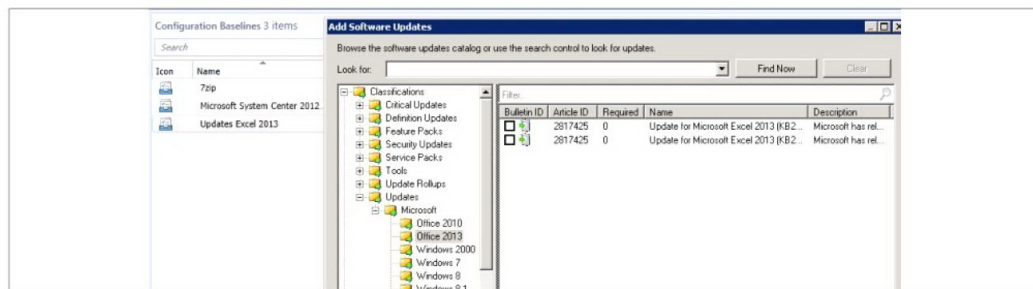


Рисунок 4. Результат импортирования Configuration Pack

Icon	Name	Status	Deployed	Date Modified	Compliance Count	Noncompliance Count	
	7zip	Enabled	Yes	4/23/2014 8:54 AM	6	1	
	Microsoft System Center 2012 Configuration Manager Server Roles	Enabled	Yes	4/23/2014 10:35 AM	2	0	
	Updates Excel 2013	Enabled	Yes	4/23/2014 8:59 AM	7	0	

Size Column to Fit
Size All Columns to Fit

- Icon
- Name
- Status
- Deployed
- User Setting
- Date Modified
- Compliance Count
- Noncompliance Count
- Failure Count
- Modified By
- Categories

В нашем случае мы сначала выбираем Configuration Item, а вторым действием добавляем обновления для MS Excel 2013 (см. рис. 3).

При добавлении Configuration Item в Configuration Baseline определяются Purpose (назначения) и находятся во вкладке Evaluation Conditions. Но при редактировании элементов необходимо знать и различать их возможные варианты присвоения, а именно:

Required (требуемый) – используется по умолчанию и означает, что, если Configuration Item не был найден, то Configuration Baseline определяется как non-compliance (несоответствующий). А иначе будет производиться процесс оценки последнего на достоверность.

Optional (опциональный) – применяется только для приложений. При наличии программы Configuration Item оценивается на соответствие, а если она отсутствует, то Configuration Baseline оценивается как неудовлетворяющий.

Prohibited (запрещенный) – присваивается тем приложениям, работу которых необходимо заблокировать. В этом случае Configuration Baseline определяется как несоответствующий, если Configuration Item будет найден на компьютере.

После создания Configuration Baseline с ним можно делать такие действия, как отключение/включение, копирование/экспорт/импорт, просмотр исходного XML, а также запуск элемента сразу, выбрав кнопку Run Summarization на панели инструментов или пункт в контекстном меню.

Пакет настроек

Configuration Pack (пакет настроек) – это созданные производителем или сообществом рекомендуемые наборы проверок параметров устройств. То есть предлагаются уже готовые упакованные (имеют MSI-расширение, а после развертывания – .cab) Configuration Item и Configuration Baseline.

Мы с вами скачаем из Microsoft Download Center [4] для сервера SCCM 2012 набор правил и настроек, пред-

назначенный для контроля следующих ролей: Management Point (точка управления), Site Server (сайт сервер) и Software Update Point (точка обновления программного обеспечения) [5].

После загрузки и установки Configuration Pack на сервере SCCM нужно импортировать CAB-файл в раздел Compliance Settings. В результате имеющиеся уже элементы дополнятся одним Configuration Baseline и четырьмя Configuration Item (см. рис. 2 и 4).

На этом подготовительные действия завершены, а заключительным этапом будет доставка единиц конфигурации на пользовательские компьютеры.

Развертывание

После того как все необходимые элементы созданы, их нужно распространить. Делается это выбором пункта Deploy из контекстного меню Configuration Baseline. В появившемся окне выбираем нужную коллекцию и устанавливаем галочку Remediate noncompliant rules when supported, которая позволяет переопределить новые параметры. Данный checkbox отвечает за автоматическое исправление конфигураций, несовместимых с требуемыми условиями, как на стационарных машинах, так и для мобильных устройств.

Отчеты

Отчеты позволяют отслеживать результаты оценки соответствия настроек на интересующих устройствах, а также помогают в выявлении имеющихся ошибок.

Если в коллекцию входит небольшое количество машин, то отчет можно сформировать на пользовательском компьютере. Для этого нужно зайти в «Панель управления → Configuration Manager», во вкладку Configuration (Конфигурация), выбрать необходимый Configuration Baseline, нажать кнопку Evaluate (Оценить), а затем View Report (Просмотреть отчет). В результате браузер отобразит исчерпывающую информацию (см. рис. 5).

Но если требуется массовый сбор данных, то удобнее перейти в рабочую область мониторинга сервера SCCM

Рисунок 5. Отчет на клиенте

Summary:									
Name	Revision	Type	Baseline Policy	Compliance State	Non-Compliance Severity	Discovery Failures	Non-Compliant Rules	Remediated Rules	Conflicting Rules
Microsoft System Center 2012 Configuration Mana...	1	Baseline		Non-Compliant	Warning	0	1	0	0
Windows Server Update Services configuration for...	1	Application Configuration	Optional	Compliant	None	0	0	0	0
Microsoft System Center 2012 Configuration Mana...	1	Application Configuration	Optional	Non-Compliant	Warning	0	1	0	0
Microsoft System Center 2012 Configuration Mana...	1	Application Configuration	Optional	Compliant	None	0	0	0	0
Microsoft System Center 2012 Configuration Mana...	1	Application Configuration	Optional	Compliant	None	0	0	0	0
NAME: Microsoft System Center 2012 Configuration Manager Management Point									
TYPE: Application Configuration									
REVISION: 1									
COMPLIANCE STATE: Non-Compliant									
NON-COMPLIANCE SEVERITY: Warning									
DESCRIPTION: Evaluates the configuration of a System Center 2012 Configuration Manager Management Point site role against Microsoft's recommended best practices.									
Non-Compliant Rules:									
Setting Name	Setting Type	Setting Description	Rule Name	Rule Description	Severity	Instance Data			
BGB firewall port is opened	Script	Verifies that the 'Big Green Button' (BGB) firewall port for this Management Point is open.	BGB firewall port is open	Firewall port for BGB feature should be open on Management Point	Warning	Expression	Current Value	Instance Source	Rule Type
						Equals Port Open	Port Closed	Property = Line[0]	Value

в разделе Monitoring → Overview → Deployments и увидеть краткую информацию и круговой индикатор (состояние) или через отчеты Monitoring → Overview → Reports → Compliance and Settings Management получить более расширенную (см. рис. 6).

Troubleshooting

Относительно архиватора и обновлений для пакета MS Excel 2013 мы поняли, что группа устройств соответствует требованиям. А что касается импортированных рекомендаций для сервера SCCM, то в отчете можем увидеть некоторые неполадки с BGB Firewall port is open (BGB – Big Green Button).

У брандмауэра порт для роли Management Point открыт в связи с условиями поставщика. Но рекомендуемая им настройка нам не важна, поэтому изменим значение правила из Equals на Not equal to во вкладке Compliance Rules (раздела Assets and Compliance → Overview → Compliance Settings → Configuration Item, в свойствах элемента Microsoft System Center 2012 Configuration Manager Management Point) и получим отчет без предупреждений и ошибок.

Еще одним способом оповещения системного администратора является журналирование. Это несколько log-файлов [6]:

- > в DCMAgent.log записывается информация об оценке и конфликтах программ, о восстановлении элементов конфигурации и приложений;
- > CAgent.log сообщает об исправлениях, настройках соответствия, обновлениях программного обеспечения и управлении приложениями;
- > в CMReporting.log помещаются данные о состоянии отправки сообщений для Configuration items;
- > DcmWmiProvider.log отвечает за сведения об инструментарии управления Windows (WMI);
- > CItaskManager.log содержит информацию о планировании задач Configuration items.

Системному администратору очень полезно иметь такую возможность, как проверка соответствия параметров

устройств предприятия. Ведь она помогает ему увидеть, как для одного компьютера, так и для групп, полную картину достоверности требуемых корпоративных политик в компании.

Так как существующие программы не всегда могут помочь добиться нужного результата (не все поддерживают работу мобильных устройств и не позволяют редактирование параметров), то функционал Compliance Settings в Configuration Manager 2012 в этом случае подходит больше и при этом не вызывает сложностей в настройке, к тому же позволяет избежать размножения уже имеющихся программных продуктов. **БОЕ**

1. Официальный сайт программы-шпиона NeoSpy – <http://ru.neospy.net>.
2. Утилита Process Monitor из набора Sysinternals – <http://technet.microsoft.com/ru-ru/sysinternals/bb896645>.
3. Документация о Compliance Settings на technet – <http://technet.microsoft.com/ru-ru/library/gg681958.aspx>.
4. Microsoft Download Center – <http://www.microsoft.com/en-us/download/details.aspx?id=30710>.
5. Описание про Configuration Pack – <http://stephanwibier.blogspot.com/2013/02/system-center-2012-configuration.html>.
6. Описание всех log-файлов – http://technet.microsoft.com/en-us/library/hh427342.aspx#BKMK_ClientInstallLog.
7. Некоторые примеры настроек Compliance Settings на блогах: разрешение рабочего стола – <http://masyan.ru/2014/02/vmware-view-sccm-2012-r2>; работа с PowerShell – <http://www.petervanderwoude.nl/post/tag/compliance-settings>; отключение обновления Adobe Flash Player – <http://blogs.technet.com/b/brandonlinton/archive/2013/03/30/how-to-create-a-configuration-item-and-baseline-to-disable-adobe-flash-player-automatic-updates.aspx>; использование SCM (GPO) – <http://www.verboon.info/2013/02/how-to-create-a-sccm-2012-sp1-configuration-baseline-with-security-compliance-manager-scm-3-0>.

Ключевые слова: соответствие, параметры, конфигурации, настройки, развертывание, правила, обнаружение.

Рисунок 6. Отчет на сервере SCCM

The screenshot shows the 'Compliance history of a configuration baseline' report in the SCCM Configuration Manager console. The report is filtered by 'Baseline Name' (7zip) and shows two entries, both with a compliance state of 'Compliant'.

Configuration Baseline Name	Configuration Baseline Revision	Device Name	User/Device	Compliance State	Severity	Last Evaluation Date and Time (UTC)
7zip	1	MININT-69A0SLB	(SYSTEM)	Compliant	None	4/23/2014 7:44:44 AM
7zip	1	SVT-BO	(SYSTEM)	Compliant	None	4/23/2014 7:42:05 AM

Настройки сети с помощью PowerShell 4.0

Microsoft заявляет, что постепенно классические консольные утилиты, к которым администраторы привыкли за долгое время, будут заменяться командами PowerShell

С выходом PowerShell 4.0 модуль Net TCP/IP [1] пополнился новыми командами, которые упростили настройку сети.

За настройку параметров интерфейса вместо `netsh interface ipv4` отвечают два команды – `New-NetIPAddress` и `Set-NetIPAddress`, синтаксис у них схож, оба могут создавать новые настройки, но второй позволяет модифицировать текущие.

```
New-NetIPAddress -InterfaceAlias Ethernet -J
-IPv4Address 192.168.0.2 -J
-DefaultGateway 192.168.0.1
Set-NetIPAddress -InterfaceAlias Ethernet -PrefixLength 24
```

Вместо `InterfaceAlias` удобнее использовать индекс `InterfaceIndex`, его легко узнать, просмотрев вывод `Get-NetIPInterface`.

Для удаления настроек IP используется команда `Remove-NetIPAddress`.

```
Remove-NetIPAddress -IPAddress 192.168.0.2
```

Для корректной работы нам понадобится настроить адреса DNS-серверов, это решается с помощью команды `Set-DNSClientServerAddress` (из состава модуля `DNSClient`):

```
Get-DnsClientServerAddress
Set-DNSClientServerAddress -InterfaceIndex 12 -J
-ServerAddresses "192.168.0.1","8.8.8.8"
```

Текущие сетевые настройки можно узнать из выводов `Get-NetIPAddress` и `Get-NetIPConfiguration` (алиас `gip`), являющихся аналогами `ifconfig`. Например, `Get-NetIPConfiguration` выводит полную информацию о сетевых настройках: IP-адреса интерфейса и шлюза и DNS. Дополнительный параметр `-Detailed` позволит получить больше данных.

Команда `route` долго использовалась для просмотра и настройки маршрутов, теперь ему на замену идет набор команд `*-NetRoute`, очень простых и понятных в использовании. Получаем текущие настройки маршрутизации и устанавливаем новый маршрут по умолчанию.

```
Get-NetRoute
New-NetRoute -DestinationPrefix "0.0.0.0/0" -J
-NextHop "192.168.0.1" -InterfaceIndex 12
```

Чтобы проверить доступность определенного узла, PowerShell предлагает два команды: `Test-Connection` и `Test-NetConnection` (сокращенно `tncc`). Первый входит в основной модуль PowerShell Management и для проверки использует ICMP (в работе напоминает обычный ping).

Второй входит в состав модуля NetTCP/IP и обладает большими возможностями, позволяя проверить доступность

определенного порта или системы с разных ПК. Например, проверим подключение к `SRV01.example.org` с двух систем:

```
Test-NetConnection -ComputerName SRV01.example.org
-source localhost, SRV02.example.org
```

Чтобы получить больше информации достаточно добавить ключ `-InformationLevel Detailed`, ключ `-TraceRoute` позволит произвести трассировку подключения (аналог `tracert`).

Для проверки, слушается ли порт, следует указать его номер (поддерживаются четыре строковых параметра – SMB, HTTP, RDP и PING). Если ввести команду без указания узла, будет проверено подключение к Интернету.

```
Test-NetConnection -Port 80 -InformationLevel Detailed
```

Количество решаемых задач может быть очень велико, для каждого условия будут оптимальны разные настройки TCP, а установки по умолчанию эффективны не всегда.

Для тюнинга TCP предназначен набор команд `*-NetTCPSetting` и `*-NetTransportFilter`.

С помощью `Get-NetTCPSetting` получаем список параметров. По умолчанию `SettingName` установлено в значение `Automatic`, и настройки пусты. Зато существенно различаются между собой `InternetCustom`, `DatacenterCustom`, `Compat`, `Datacenter` и `Internet`, назначение которых понятно и без объяснений. Но редактировать можно только профили `InternetCustom` и `DatacenterCustom`. Какие подключения будут использовать профиль, можно выяснить, выполнив `Get-NetTCPConnections`.

Включим расширение `IP ECN (Explicit Congestion Notification)`, явное уведомление о перегруженности, см. RFC 3168) и протокол `DCTCP (Data Center TCP)` для `DatacenterCustom`.

```
Set-NetTCPSetting -SettingName DatacenterCustom -J
-ECNCapability Enabled -CongestionProvider DCTCP
```

Чтобы применить параметр TCP к порту с номером или диапазоном IP-адреса назначения, надо создать транспортный фильтр с помощью команды `New-NetTransportFilter`.

```
New-NetTransportFilter -SettingName DatacenterCustom -J
-DestinationPrefix 172.16.0.0/16
```

Дополнительно можно задать локальный и удаленный порты. **BOX**

1. Описание команд модуля Net TCP/IP – <http://technet.microsoft.com/en-us/library/hh826123.aspx>.

Подготовил Сергей Яремчук



Визитка

АЛЕКСЕЙ БЕРЕЖНОЙ, независимый консультант, системный архитектор, специалист по системам виртуализации и резервного копирования, alexey.berezhnoy@tech-center.com

Не теряя управления

Часть 2. Обзор средств удаленного управления

Продолжаем разговор о создании отказоустойчивых решений, предотвращающих потерю управления сервером в критичной ситуации

После выхода статьи «Не теряя управления» [1] я получил письмо с просьбой подробнее рассказать о средствах сохранения возможности управления сервером в любой ситуации. Один из таких методов – правильный выбор инструментов удаленного доступа и создание соответствующей среды для их применения.

Краткий обзор средств удаленного управления

В статье «Не теряя управления» я привел пример инцидента, когда сервер потерял управление из-за излишне «прожорливого» процесса резервного копирования. Все попытки удаленно «достучаться», используя обычные средства управления Windows Server: Remote Desktop Connection, MMC-консоль и консоль управления системой резервного копирования, потерпели неудачу. И только старый добрый сервис Telnet (который, кстати, обычно выключен по умолчанию) спас ситуацию. В итоге, используя интерфейс командной строки, удалось остановить процесс резервного копирования, тем самым освободив ресурсы сервера.

Итак, мы убедились, что необходимо иметь минимум два средства удаленного управления своими серверами.

При этом одно из них должно быть реализовано как интерфейс командной строки на случай, если из-за серьезной деградации производительности использовать графические среды не представляется возможным.

Существует два направления для решения подобных задач: аппаратное и программное. В принципе это не новость, такого рода дуализм характерен для большинства областей, связанных с вычислительной техникой.

Приобретение аппаратных и специализированных программных средств удаленного управления можно рассматривать как своего рода инвестиции в развитие ИТ-инфраструктуры компании. Когда предприятие еще небольшое, эти меры кажутся избыточными, но в дальнейшем такой подход приобретения средств управления «на вырост» окажется достаточно разумным и предусмотрительным. При этом не стоит забывать, что удаленное управление и резервное копирование часто стоят рядом. Никто не гарантирован от запуска тяжелых процессов задач вне «окна бэкапа»,

в том числе и по случайным причинам. И возможность получить управление сервером в любое время при любых обстоятельствах является одной из важных деталей при построении надежной ИТ-системы.

Но бывает, что компания находится в сложном финансовом положении, не готова делать значительные вложения, которые понадобятся только в отдаленной перспективе. Поэтому существуют облегченные решения, которые позволяют системным архитекторам, инженерам и администраторам решить часть проблем обеспечения управляемости серверов в форс-мажорных обстоятельствах на уровне малых затрат.

Я решил построить материал по этим вопросам следующим образом. Вначале мы рассмотрим аппаратные инструменты удаленного управления, наиболее независимые от операционной системы и программного обеспечения сервера. Затем остановимся на штатных средствах, предоставляемых ОС, как наиболее доступных, по принципу «всегда под рукой».

И, наконец, коснемся программных средств от сторонних разработчиков, которые в сочетании с выделенной сетью для управления серверами на базе дешевого сетевого оборудования помогут организовать малобюджетное решение для компаний, которые сейчас не могут или не хотят тратить деньги на ИТ.

Аппаратные средства удаленного управления

Аппаратные средства удаленного управления также можно разделить на две основные категории. В первую входят специальные модули, встраиваемые в серверное оборудование известных брендов, например, HP Integrated Lights-Out (iLO) [2] или [3]. Данная концепция базируется на средствах Intelligent Platform Management Interface (IPMI) – интеллектуальный интерфейс управления платформой [4].

Обычно покупателю оборудования предлагается несколько функциональных уровней.

Базовый – в нем доступен очень ограниченный перечень операций, например, контроль над системой охлаждения, удаленная перезагрузка/включение/выключение сервера.



Сохранение управляемости сервера в критической ситуации необходимо при обеспечении отказоустойчивости системы

Стандартный – в этом случае обычно добавляется возможность использовать интерфейс командной строки, также улучшена система мониторинга. Иногда в этот же уровень включают обновление прошивки, настройку CMOS Setup [5] и другие интересные вещи.

Расширенный – в нем уже присутствует полноценное управление серверов (обычно через Java-консоль по HTTPS-протоколу, используя обычный интернет-браузер). Кроме того, может быть добавлена функция подключения удаленных устройств, используя физические приводы и файлы образов на компьютере пользователя (администратора), управления RAID-контроллером и так далее. Фактически реализуется максимально функциональный уровень управления сервером.

Соответственно первичный, самый «слабый», уровень для покупателей оборудования предоставляется бесплатно, а за расширенные возможности придется заплатить, приобретая специальную лицензию или даже дополнительный аппаратный модуль.

Имея такую выделенную локальную сеть для подключения к аппаратным средствам управления, нетрудно создать независимое отказоустойчивое решение для работы с серверным оборудованием и ПО в ситуациях, когда сервер «в глухую» занят тяжелым процессом или попросту перестал отвечать из-за сбоя.

К другой группе относятся решения на базе IP KVM [6]. Это аппаратные консоли для переключения одного комплекта из клавиатуры, монитора и мышки к нескольким серверам, но со встроенным интерфейсом управления по IP-сети. Пользователю предоставляется веб-интерфейс, имитирующий физическую консоль сервера. А для реализации возможности удаленной «холодной» перезагрузки серверов можно использовать управление портами блока бесперебойного питания. Отключая подачу напряжения на том или ином выходном порту, можно произвести перезапуск сервера временным отключением питания (при условии, что BIOS сервера позволяет выбрать соответствующую опцию для включения сервера сразу после подачи питания). Несмотря на гораздо меньший функционал такого подхода

по сравнению с IPMI, подобное решение подходит для большинства оборудования, в том числе и для полюбившихся российскому бизнесу «самосборных вариантов», когда «сервер» собирается вручную, чаще всего из комплектующих для рабочих станций.

Вне зависимости от типа выбранного аппаратного решения его можно охарактеризовать словами: «дорого, но удобно и надежно». В то же время высокий уровень затрат подталкивает специалистов и руководителей в области ИТ к рассмотрению других вариантов.

Штатные программные средства удаленного управления

Если говорить о средствах удаленного управления, встроенных в ОС, первый и самый очевидный факт заключается в прямой зависимости от специфики их «родной операционной системы». Поэтому все достоинства и недостатки – завышенные системные требования, неэффективная работа с тем или иным программным или аппаратным модулем – влияют на работу этих инструментов.

Если говорить о системах Windows, то здесь стоит выделить два основных инструмента: Remote Desktop Connection (Подключение к рабочему столу) и старый добрый Telnet. В то же время штатные средства ОС от Microsoft являются зависимыми от других служб Windows.

Например, Remote Desktop Connection зависит от службы «Удаленный вызов процедур (RPC) (Remote Procedure Call (RPC)» [8], а Telnet Server, помимо данной службы, зависит еще и от служб NT LM (NT LM Security Support Provider) [9] и TCP/IP Protocol Driver (IPSEC driver).

Таким образом, проблемы с той же службой RPC (например, из-за утечки памяти или запуска особенно ресурсоемкого процесса) могут вызвать «эффект домино» и породить недоступность служб удаленного управления сервером, с помощью которых системный администратор по идее и должен решить проблемы.

Итак, круг замкнулся по принципу, описанному в одном из законов Мерфи: «Ключ от срочной медицинской комнаты хранится в срочной медицинской комнате». Чтобы выйти

из данной щекотливой ситуации, нужно иметь дополнительные, «внештатные», средства управления серверами.

В UNIX-системах таких проблем обычно не возникает, там по умолчанию используются стандартный протокол и соответствующий сервис SSH, который имеет низкие требования к ресурсам и неплохую защиту с использованием асимметричных ключей шифрования. А при использовании SFTP (расширения протокола SSH-2) можно осуществлять копирование файлов.

В Windows-системах в ряде случаев возможно использование SSH-серверов сторонних разработчиков, об этом речь пойдет ниже. Но в Windows-системах нет такой встроенной «роскоши». Что же касается Telnet, то и его можно использовать только внутри периметра защищенных локальных сетей.

Приобретение аппаратных и специализированных программных средств удаленного управления можно рассматривать как своего рода инвестиции в развитие ИТ-инфраструктуры компании

Примечание. Еще есть возможность использовать PowerShell, но я бы не назвал это подходящим инструментом для работы в сложных ситуациях. Во-первых, из-за достаточно ресурсоемкой реализации на базе вызова функций .NET. Во-вторых, крайне неудобно набирать длинные команды, состоящие из множества символов, с не менее длинными параметрами, которые практически невозможно запомнить наизусть и нужно постоянно подглядывать в справочник. Довольно часто приходилось наблюдать, как ситуация меняется к худшему гораздо быстрее, нежели системный администратор успевает набрать требуемую команду в PowerShell. И очень хотелось бы, чтобы компания Microsoft наконец повернулась лицом к системным администраторам и системным инженерам и предложила более удобный вариант работы с командной строкой. Например, как у Марка Руссиновича в его утилитах Sysinternals.

Поэтому крупные предприятия предпочитают приобретать дорогое оборудование от известных брендов со встроенными средствами управления, как, например, HP Integrated Lights-Out (iLO), о которых было рассказано в предыдущем разделе.

Как я уже говорил, такие решения обходятся значительно дороже, чем обычное серверное оборудование от небольших компаний. При этом для полноценного использования придется купить отдельную не дешевую лицензию (как в случае с тем же iLO от HP).

Примечание. Помимо основных средств, основанных на протоколах RDP и Telnet, существует довольно большая группа инструментов более узкого назначения, с помощью которых можно осуществлять удаленное управление

в рамках той или иной задачи. Например, MMC-консоль или Server Manager для управления сервером позволяет останавливать и запускать службы, настраивать параметры безопасности и так далее.

На фоне современных международных событий и возможных санкций ЕС и США по отношению к России, вполне вероятно, могут возникнуть перебои с поставками оборудования известных брендов. В то же время на территории Российской Федерации существуют компании, которые сами занимаются сборкой серверов, но при этом не обладают такими технологиями, как iLO.

Конечно, можно использовать и оборудование IP KVM для удаленного доступа к физической консоли сервера. Но такие решения достаточно дороги и плохо масштабируемы. Например, если есть переключатель IP KVM на восемь портов, то при покупке девятого сервера придется покупать и устанавливать новый переключатель IP KVM. Поэтому использование такой технологии доступно далеко не всем компаниям.

Альтернативой использованию аппаратных средств является применение дополнительной сети управления, которая может быть организована на недорогом сетевом оборудовании только для удаленного доступа к серверной инфраструктуре. Но все равно вопрос о программных средствах остается открытым.

Программные средства удаленного управления от третьих фирм

Сегодня существует довольно много сторонних решений от независимых производителей ПО. Достаточно вспомнить такие интересные в своем роде продукты, как pcAnywhere от Symantec [10], DameWare Mini Remote Control [11] и другие. Наиболее предпочтительным направлением, на мой взгляд, является применение либо полностью бесплатных решений на базе открытого протокола VNC, либо платных решений от российских производителей. Например, решение компании Radmin работает на собственном протоколе в рамках стека TCP/IP. (Продукты российских компаний стоит выбирать, не только вдохновившись призывами поддержать отечественного производителя, но также из-за риска дальнейшей изоляции России, в том числе и со стороны иностранных разработчиков ПО.)

Решений на основе VNC для Windows-систем существует достаточно много. Я в свое время использовал программный продукт UltraVNC, который является полностью бесплатным (UltraVNC is free for ALL usage [13]). В число предоставляемых возможностей входит использование шифрования при доступе к удаленной системе, интеграция с Active Directory, передача файлов и даже встроенный чат между пользователями. В принципе для поддержания работы персональных компьютеров офисных пользователей в обычной обстановке UltraVNC в большинстве случаев достаточно. Существует и механизм развертывания UltraVNC через групповые политики (GPO) [14].

Минусом является тот факт, что VNC – это все-таки достаточно ресурсоемкий протокол, требующий неплохих каналов передачи данных и приличного серверного оборудования, поэтому он не слишком подходит для решения проблем, связанных с резким возрастанием нагрузки на сервер.

Концепция решений на базе Open Source (к числу которых принадлежат и продукты на основе VNC), к сожалению, пока еще не понята российским обществом. Написание отчетов об ошибках (bug-reports) для дальнейшего устранения проблемы разработчиками и общение с сообществом (Community) разработчиков и пользователей того или иного открытого продукта почему-то до сих пор вызывают сложности у многих системных администраторов и других ИТ-специалистов. К тому же сама мысль о том, что вне зависимости от «бесплатности» продукта его поддержка может и должна оставаться платной, что на нее необходимо выделять деньги, вгоняет руководство большинства российских компаний в полный ступор. Поэтому часто бывает проще купить готовое решение, нежели адаптировать бесплатный продукт собственными силами.

Стоит также обратить внимание на чрезмерную «опеку» российского бизнеса государством. Всеобъемлющая запретительная политика «всего и вся» вполне может привести к печальным последствиям, когда, встретив при проверке незнакомую программу, представители всевозможных проверяющих структур могут попытаться забрать компьютеры «на экспертизу». Наличие документа о покупке программного продукта у российской компании снижает подобные риски.

Если говорить о Radmin, то этот продукт в качестве примера я выбрал по нескольким причинам.

Как я уже писал выше, это отечественный продукт, который признается российскими контролирующими органами и изъавлен от рисков, связанных с экономическими и политическими санкциями.

Данный продукт не требователен к ресурсам сервера, не нуждается в специальном широком канале. Это достигается за счет специальной оптимизации для соединений с низкой пропускной способностью: имеется возможность понижения цветности изображения, а также применяются интеллектуальные алгоритмы компрессии изображения. Radmin отслеживает и передает только изменения удаленного экрана, при этом использует эффективный метод компрессии, что позволяет повысить скорость работы.

Конечно, это не означает, что, купив Radmin, вообще не нужно создавать выделенную сеть для управления серверами. Но то, что он может работать на медленном канале, в случае высокой загрузки весьма повышает его ценность в качестве аварийного средства обеспечения доступности серверов.

При этом функциональность данного продукта весьма впечатляет. Пользователю предоставляются следующие возможности:

- > Безопасный обмен файлами с функцией «докачки».
- > Поддержка переключения сессий пользователей в Windows 8, Windows 7, Vista и Windows XP.
- > Адресная книга с древовидной структурой, drag-and-drop для записей и папок.
- > Встроенный сканер серверов Radmin.
- > Режим Telnet.
- > Удаленное выключение компьютера.
- > Автоматизированная система удаленного развертывания серверов.
- > Текстовый и голосовой чаты.
- > Поддержка нескольких одновременных соединений.

Лицензия приобретается только на сервер, к которому осуществляется удаленный доступ. Стоимость одной лицензии сегодня – 1250 рублей и ниже, при этом срок действия лицензии не ограничен. При этом клиентская часть бесплатна и может быть установлена на неограниченное количество компьютеров, с которых осуществляется управление.

По поводу безопасности могу сказать, что продукт неплох защищен. В частности, реализованы следующие средства:

- > Шифрование соединения с учетом современных требований к безопасности.
- > Интеллектуальная защита от подбора пароля, включающая систему задержек и блокировку подозрительных IP-адресов.
- > IP-фильтрация, которая позволяет разрешить доступ только с определенных IP-адресов или их диапазонов.
- > Запрос на подтверждение входящего соединения.

Альтернативой использования аппаратных средств является применение дополнительной сети управления

Наверное, стоит более подробно остановиться на шифровании трафика.

Radmin работает в режиме защиты данных, при котором все передаваемые данные, изображения экрана, перемещение курсора и сигналы клавиатуры надежно защищены с длиной ключа в 256 бит. Секретный ключ генерируется случайным образом для каждого подключения. Для аутентификации пользователей в Radmin может быть использована либо система безопасности Windows с поддержкой Active Directory и протокола Kerberos, либо собственная система безопасности Radmin с индивидуальными правами доступа для каждого пользователя и защищенной аутентификацией по логину и паролю.

Обновление версий Radmin происходит одновременно с обновлением операционных систем Windows. Партнерство с Microsoft позволяет разработчикам Radmin обеспечивать совместимость программы с новыми версиями Windows еще до их официального выхода.

Из дополнительных плюсов хочется отметить крайнюю неприхотливость данной системы. В первую очередь это минимальные требования к работе операционной системы. Если операционная система запустится, Radmin соответственно тоже.

Как я уже писал выше, существует специальный инструмент – программа Radmin Deployment, с помощью которой можно создать MSI-пакет с заданными настройками и произвести его массовую установку на компьютерах локальной сети.

Инструмент имеет удобную консоль и позволяет быстро развернуть серверные модули удаленно (что бывает довольно полезно при передаче услуг на аутсорсинг). Подробнее о данной функции можно прочесть в стандартной документации на сайте разработчика.

Данная статья была бы неполной, если бы я не вспомнил еще об одной замечательной возможности – использовании независимых реализаций SSH-серверов для операционных систем от Microsoft Windows. В свое время хорошо себя зарекомендовал продукт FreeSSHd [15], который я представлял системным администраторам в своей статье [16]. В качестве возможности удаленного управления Windows-серверами по защищенному протоколу через инструмент командной строки он весьма неплох.

Графические средства хорошо подходят для упрощения процесса администрирования тем или иным сервисом

Кратко о UNIX-системах

Не стоит наивно полагать, что проблемы потери управления возможны только в Windows-системах. В удивительном мире UNIX также случаются неприятные ситуации, когда тот или иной сервер временно оказывается недоступен по разным причинам. Рекомендации для страховки от подобных случаев примерно те же: организовать выделенную сеть управления, по возможности использовать аппаратные средства или, на худой случай, программные средства, рассчитанные на работу под высокой нагрузкой.

Графические средства хорошо подходят для упрощения процесса администрирования тем или иным сервисом. Из таких инструментов для удаленного управления можно порекомендовать инструмент Webmin, который позволяет через веб-интерфейс управлять сервером на расстоянии.

Еще одним из плюсов данной системы является возможность делегирования полномочий управления каким-либо сервисом в рамках Webmin [17]. У этого хорошо зарекомендовавшего себя «ветерана» есть и конкуренты, например, Ajenti [18]. В принципе неважно, какой именно сервис используется, главное, чтобы он удовлетворял специфическим требованиям ИТ-инфраструктуры и мог работать в условиях высокой нагрузки. Тем самым реализуется основное правило: не менее двух систем удаленного управления на случай отказа одной из них.

Из инструментов командной строки пока что лидирует SSH, но может оказаться полезной возможность удаленной «холодной» перезагрузки, например, через управление блока бесперебойного питания. Это необходимо на случай, если сервер окажется недоступен. При этом необходимо отдавать себе отчет, что файловые системы UNIX-систем более критичны к несанкционированным перезагрузкам, поэтому подобные «простые решения» могут сами оказаться источником будущих проблем.

О системах виртуализации

Использование систем виртуализации, с одной стороны, может помочь решить проблему, при которой гостевая машина теряет управление. Стандартные средства управления системой виртуализации сами по себе являются

хорошим резервным инструментом на случай недоступности штатных средств управления ОС VM. С другой стороны, потеря управления хоста может привести к плачевным результатам. Поэтому ответственность ИТ-персонала, обслуживающего виртуальные системы, гораздо выше, и относиться к проблемам стоит гораздо серьезнее. Поэтому необходимо тщательно рассчитывать максимальную нагрузку на хост-серверы и не превышать допустимые значения.

Сохранение управляемости сервера в критической ситуации крайне необходимо при обеспечении отказоустойчивости системы в целом. К счастью, существует достаточно много средств, позволяющих не допустить потери управления. Но вне зависимости от того, какие отличные средства применялись для этого, следует помнить, что основная задача – обеспечить нормальный режим работы и не доводить ситуацию до критического состояния. Самый лучший выход из кризисной ситуации – недопущение ее в принципе. **БОР**

1. Бережной А. Не теряя управления. // «Системный администратор», №5, 2014 г. – С. 22-26 (<http://samag.ru/archive/article/2685>).
2. Описание интерфейса iLO на сайте HP – http://h17007.www1.hp.com/us/en/enterprise/servers/management/ilo/index.aspx#_U3NqsCgfPZM.
3. Описание инструмента IMM – http://en.wikipedia.org/wiki/IBM_Remote_Supervisor_Adapter.
4. Статья о IPMI – <http://www.etegro.ru/articles/ipmi-bmc>.
5. Краткое знакомство с CMOS Setup – http://on-line-teaching.com/bios/01_cmos.html.
6. Материал об IP KVM – http://interface31.ru/tech_it/2011/02/polnyj-kontrol-pereklyuchatel-kvm-over-ip.html.
7. Информация о Terminal Service на Microsoft TechNet – <http://technet.microsoft.com/ru-ru/library/cc770412.aspx>.
8. Описание службы «Удаленный вызов процедур (RPC) (Remote Procedure Call (RPC))» – <http://www.oszone.net/display.php?id=2590>.
9. Описание службы «Поставщик поддержки безопасности NT LM (NT LM Security Support Provider)» – <http://www.oszone.net/display.php?id=2550>.
10. Домашняя страница проекта pcAnywhere (Symantec) – <http://us.norton.com/symantec-pcanywhere>.
11. Сайт продукта DameWare Mini Remote Control – <http://www.dameware.com/products/mini-remote-control/product-overview.aspx>.
12. Страница проекта RemotelyAnywhere – <http://remotelyanywhere.com/template.asp?page=home>.
13. Официальный сайт проекта UltraVNC – <http://www.uvnc.com>.
14. Описание инструмента развертывания UltraVNC посредством групповых политик – <http://greendail.ru/node/400>.
15. Сайт проекта FreeSSHd – <http://www.freesshd.com>.
16. Бережной А. «DOWNLOAD» – сисадминские штучки». Сайт журнала «Системный администратор» – <http://samag.ru/uart/more/51>.
17. Сайт проекта Webmin – <http://www.webmin.com>.
18. Домашняя страница проекта Ajenti – <http://www.ajenti.com>.

Ключевые слова: удаленное управление, VNC, Radmin, SSH, RDS.

Набор утилит dttools

Управляем множеством UNIX-систем

В подчинении сисадмина обычно находится не один, а несколько серверов. Давать команды каждому по отдельности неэффективно, на помощь приходят специализированные инструменты — Fabric, pyDSH, dttools и другие

Первые два основаны на Python и при необходимости вполне могут заменить более сложные инструменты; dttools (Distributed tools [1]) написан на bash и по идеологии и подходу ближе к классическому shell, а поэтому очень прост в понимании и использовании. При необходимости dttools легко расширяется. В репозиториях дистрибутивов Linux dttools обычно не встречается, но его установка проста.

```
$ sudo apt-get install git
$ git clone https://github.com/ajdiaz/dttools.git
$ cd dttools
$ sudo cp dt dtstatus /usr/bin
$ sudo cp -r lib /usr/lib/dttools
$ sudo cp -r doc/man/ /usr/share/man/man1
```

Все операции выполняются с помощью bash скрипта dt, синтаксис выглядит просто:

```
dt pattern command arguments
```

Шаблоны представляют собой фильтр-выборку целевых хостов. Сами узлы описываются в базе, по умолчанию это файл ~/.dtdb, но с помощью параметра -D/--db можно указать любой другой файл (или, как вариант, изменить значение переменной DTOOLS_DB=~/.dtdb в самом скрипте). Каждый узел описывается в отдельной строке, после которой указываются тэги, которым он будет соответствовать.

```
localhost www smtp mysql
host1 www mysql wp
host2 www pg wp
```

Пример самый простой, на самом деле тэгов и узлов может быть много, и, таким образом, можно описать все возможные варианты для подачи команд. Как и принято в UNIX, названия регистрозависимы, то есть mysql и MySQL — это разные тэги.

Чтобы просмотреть список узлов для определенного тэга, необходимо использовать параметр list:

```
$ dt tag:www list
```

Возможна выборка узлов с несколькими тэгами. Например, выберем все хосты, у которых есть тэги mysql или wp:

```
$ dt tag:mysql+tag:wp list
```

Или нам нужны только узлы, у которых присутствуют оба тэга одновременно:

```
$ dt tag:mysql:tag:wp list
```

Все шаблоны описаны файлами в /usr/lib/dttools/patterns (в ~/.dttools/patterns шаблоны может создавать сам пользователь). Кроме указанных, есть также и операция отрицания, например, все тэги, кроме wp, можно отобрать с помощью tag:wp:not::, и другие шаблоны all (все), nil (по сути, пустышка), ec2 (узлы AWS EC2), out (внешний URL), и использование регулярных выражений.

Выборку возможно производить с помощью нескольких шаблонов. Но результат лучше вначале протестировать с помощью list или специального ключа -p/--pretend.

Для удобства можно использовать три парсера (dthost, dtstatus, dtcompact), выводящие результат в разной форме.

```
$ dt exp:.* list | dthost
```

Узлы в ~/.dtdb можно заносить вручную, но, когда список большой, это становится неудобно. Здесь проще использовать команду add. Добавим host3 с тэгом mysql:

```
$ dt sys:host3 add mysql
```

Операции производятся через SSH. Вначале следует подключиться по ssh к удаленной системе, и узел будет автоматически добавлен ~/.ssh/known_hosts. Это можно сделать и самому:

```
ssh-keyscan host >> ~/.ssh/known_hosts
```

Все должно работать автоматически. Но здесь есть нюансы. По умолчанию для аутентификации используется открытый ключ, а не логин/пароль, а поэтому в зависимости от ситуации будет выдаваться ошибка. Удобнее скопировать открытый ключ на удаленные системы, это можно сделать вручную или с помощью команды ssh-copy-id. Но в обычном режиме это вряд ли получится, поэтому следует использовать интерактивный (-i), вводя пароль к каждому узлу:

```
$ dt -i exp:.* ssh-copy-id ~/.ssh/id_dsa.pub
```

Здесь есть тонкость, не описанная в документации, — подключение производится с помощью текущей учетной записи, и как-либо указать иную нельзя. Но выход найти просто — узлы базы следует прописывать в стиле ssh, как user@host. Команды, которые может выполнять dttools, — это именно «внутренние» команды, а не команды ОС. Хотя для удобства они созвучны **EOF**

1. Сайт dttools — <http://ajdiaz.me/dttools>.

Подготовил Сергей Яремчук



Визитка

ИГОРЬ ОРЕШЕНКОВ, ОАО «Банк БелВЭБ»,
заведующий сектором информационно-технической
поддержки, iharsw@tut.by

Инфокиоск своими руками

Как подготовить «общественный компьютер» для работы в роли тонкого клиента и защитить его от нецелевого использования

Обзор устройств самообслуживания

С ростом информатизации общества увеличивается число услуг, которые можно получить удаленно. С помощью персонального компьютера, планшета или смартфона можно посмотреть афишу кинотеатра, узнать расписание авиарейсов и забронировать номер в гостинице.

Однако, чтобы разместить свои услуги ближе к клиентам, организации торговли и сервиса, транспортные компании и финансовые учреждения устанавливают в общественных местах устройства самообслуживания. Для получения наличных или оплаты коммунальных платежей можно воспользоваться банкоматом или платежно-справочным терминалом, в инфокиоске на вокзале – уточнить график следования поездов и приобрести билет.

Несмотря на все многообразие устройств самообслуживания, их конструкции очень похожи. В одном корпусе монтируется сенсорный монитор, чековый принтер, специализированное оборудование (кардридер, клавиатура ввода ПИН-кодов, устройство выдачи банкнот и др.) и компьютер серийного образца, дополнительно оснащенный контроллером специального оборудования. Управляет всем этим аппаратным комплексом специализированное программное обеспечение, работающее в среде операционной системы Windows, UNIX, Linux или даже DOS.

И спрос на такие устройства растет. Магазины увеличивают свою привлекательность для покупателей, размещая информационные киоски с рекламными материалами, сведениями о новинках и каталогами товаров. В банках и общественных учреждениях внедряются системы управления очередями. В России уверенность в росте рынка инфокиосков подкрепляется государственными программами [1].

Компьютер общего пользования

Разработка устройства самообслуживания – сложный наукоемкий процесс. В то же время для работы с каталогом книг в библиотеке достаточно возможностей обычного компьютера. На таком же компьютере можно организовать просмотр трейлеров фильмов-новинок в службе видеопроката, разместить каталог товаров в магазине. Для этого достаточ-

но запустить на компьютере веб-браузер и предоставить доступ к сайту с необходимой функциональностью.

Можно, конечно, установить в доступном месте обычный ПК. Но среди посетителей не все пользователи одинаково аккуратны, кто-то неумышленно может нарушить работу системы, а иной попытается использовать такой компьютер для взлома сети учреждения. Чтобы этого не произошло, при подготовке компьютера общего пользования нужно предпринять специальные меры предосторожности [2, 7].

Нам нужен компьютер с веб-браузером в качестве тонкого клиента. Реализовать это можно почти на любой современной ОС. Тогда почему бы не рассмотреть бесплатные варианты? Все чаще можно услышать разговоры о Chromium OS – операционной системе, базирующейся на кодах Chrome OS и потенциально являющейся идеальным кандидатом для нашей цели. Однако на сегодняшний день она еще слишком молода, и даже ее компиляция для конкретного компьютера не является тривиальной задачей.

Установка программного обеспечения

Поскольку «старый друг лучше новых двух», в очередной раз обратимся к Linux, а точнее, к дистрибутиву Ubuntu, чтобы заручиться поддержкой обширного сообщества поклонников этой ОС. После изучения сообщений на форумах по нашей проблеме становится ясно, что в качестве основы для инфокиоска лучше использовать либо серверную редакцию (и установить на нее недостающее программное обеспечение вручную), либо легковесный дистрибутив Lubuntu на основе графической оболочки LXDE [3]. В этой статье рассматривается второй вариант.

Чтобы установка системы прошла успешно даже на морально устаревшем компьютере, будем использовать «альтернативный» дистрибутив Lubuntu 14.04 для процессоров x86 [4]. Руководств по установке написано много, не буду описывать подробно. В дальнейшем мы предполагаем, что:

- > в ходе установки будет создана учетная запись `kioskadmin`, под которой будем выполнять все манипуляции и работу;
- > доменное имя веб-ресурса – www.webserver.com.

В интернет-дискуссиях о настройке компьютеров общего пользования чаще всего обсуждаются браузеры Chromium и Firefox. По-видимому, пользователи голосуют за свободную BSD-лицензию и открытые коды Chromium, сравнивая ее с закрытой проприетарной (хотя и бесплатной) лицензией Google Chrome. Несмотря на то что браузеры Chromium и Firefox построены на разных движках, в практическом использовании между ними сложно найти принципиальные отличия – сказывается высокая степень стандартизации. Описанные в статье меры защиты системы универсальны, а поскольку адаптация Chromium требует больше манипуляций, то именно его настройку опишем более подробно. О работе Firefox в режиме инфокиоска скажем несколько слов в конце статьи.

Установим недостающее ПО – сервер SSH и браузер Chromium (Firefox присутствует на дистрибутивном CD). Для этого достаточно выполнить:

```
$ sudo apt-get install openssh-server chromium-browser
```

Если доступа к Интернету нет, потребуется заранее скачать пакеты с зеркала официального репозитория Ubuntu [5]. Счастливые обладатели установленной на альтернативном компьютере системы Lubuntu могут получить ссылки на необходимые файлы с помощью команды (если пакеты не установлены):

```
$ sudo apt-get -y --print-uris install openssh-server chromium-browser
```

Скачиваем их, переносим на нужный компьютер и ставим:

```
$ sudo dpkg -i ./*.deb
```

Другой вариант – создание репозитория на CD вручную (dpkg-scansnapackages) или с помощью AptOnCD и последующий перенос на другие системы.

Теперь система готова к превращению компьютера в инфокиоск.

Защита сети

Компьютер общего пользования отличается от других компьютеров организации тем, что в большей степени приходится защищать не его от сетевых атак, а сетевые ресурсы от пользователей этого компьютера. В идеале было бы неплохо сконфигурировать активное сетевое оборудование так, чтобы инфокиоск и необходимые ему ресурсы попали в отдельный сегмент сети VLAN. Помимо этого, настроим брандмауэр Netfilter таким образом, чтобы:

- > инфокиоском можно было удаленно управлять по протоколу ssh;
- > инфокиоск имел доступ к веб-серверам с демонстрируемым сайтом;
- > инфокиоск мог использовать службы DNS и NTP;
- > блокировать весь трафик IPv6.

Для этого требуется создать файл /etc/network/if-up.d/iptables-rules [6] со следующим содержимым:

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
```

```
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p tcp -m conntrack --ctstate NEW -m tcp -dport 22 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
-A OUTPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A OUTPUT -p icmp -j ACCEPT
-A OUTPUT -p tcp -m conntrack --ctstate NEW -m tcp -dport 53 -j ACCEPT
-A OUTPUT -p udp -m conntrack --ctstate NEW -m udp -dport 53 -j ACCEPT
-A OUTPUT -p udp -m conntrack --ctstate NEW -m udp -dport 123 -j ACCEPT
-A OUTPUT -d www.webserver.com -p tcp -m conntrack --ctstate NEW -m multiport --dports 80,443
-A OUTPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

Применить настройки командой:

```
$ sudo /etc/network/if-up.d/iptables-rules
```

Ограничение доступа к сайту www.webserver.com в представленном виде будет работать только в том случае, если инфокиоск имеет к серверу [webserver.com](http://www.webserver.com) маршрутизируемый доступ. Разрешение доменного имени будет выполнено в момент инициализации фильтра (например, при загрузке компьютера), и в ходе работы станет использоваться полученный на этом этапе IP-адрес (или IP-адреса).

Когда же сайт доступен через прокси-сервер (NGINX, squid или другое решение), в правилах пакетного фильтра вместо веб-сервера нужно указать IP-адрес прокси-сервера.

Если у инфокиоска не будет доступа в Интернет, в файле /etc/ntp.conf можно указать NTP-сервер организации, а в файле /etc/resolv.conf – доступный DNS-сервер.

Настройка учетной записи

Для повседневной работы с инфокиоском нужна ограниченная учетная запись, входящая в единственную системную группу nopasswdlogin, для того, чтобы при включении компьютера происходил автоматический вход в систему. Создаем пользователя kioskuser:

```
$ sudo useradd -m -U -G nopasswdlogin kioskuser
```

Назначаем ему новый пароль:

```
$ sudo passwd kioskuser
```

Настройка графического сеанса

Нам нужно добиться, чтобы после включения компьютера автоматически происходил запуск браузера. Упрощенно процедура загрузки ОС выглядит, как показано на рис. 1.

Ядро запускает процесс init, который в конечном итоге включает графическую подсистему и менеджер дисплея. После успешной авторизации запускается сеанс, соответствующий учетной записи пользователя. Как правило, он представляет собой оконный менеджер с окружением рабочего стола. Для наших целей потребуется создать сеанс, в котором вместо окружения рабочего стола будет запускаться единственное доступное пользователю приложение – интернет-браузер.

Вариант 1 (популярный). Выделенный сеанс

В подавляющем большинстве обсуждений способа построения инфокиоска на базе различных дистрибутивов Ubuntu предлагается на уровне менеджера дисплея создать специальный сеанс, в котором обеспечить запуск необходимых для работы киоска приложений [7].

Создадим файл сеанса /usr/share/xsessions/kiosk.desktop со следующим содержанием:

```
[Desktop Entry]
Name = Kiosk
Comment = Kiosk Mode
Exec = /usr/bin/kiosk
Type = Application
```

Его задача – при выборе пользователем сеанса Kiosk выполнить сценарий /usr/bin/kiosk. А в сценарии мы так пропишем монополюсный запуск браузера chromium-browser:

```
#!/bin/sh
/usr/bin/openbox &
/usr/bin/chromium-browser
```

Файл со сценарием нужно сделать исполняемым:

```
$ sudo chmod a+x /usr/bin/kiosk
```

Из текста сценария видно, что перед запуском браузера в фоновом режиме стартует оконный менеджер Openbox. Справедливости ради можно отметить, что chromium-browser может работать и без оконного менеджера. Но в этом случае не срабатывает определение размеров графического экрана для полноэкранного режима (что решается указанием нужных значений через параметр командной строки браузера), и графический указатель имеет вид «креста», а не стрелки. У браузера Firefox без оконного менеджера нарушается функционирование элементов управления – например, на сайте не работают выпадающие списки.

Рисунок 1. Загрузка ОС Ubuntu

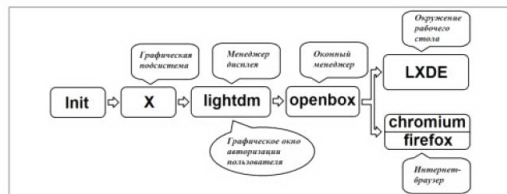


Рисунок 2. Выбор графического сеанса

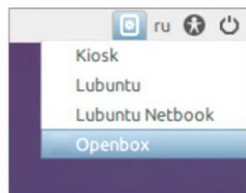


Рисунок 3. Запуск эмулятора терминала в сеансе Openbox



Получить полнофункциональную систему без оконного менеджера не удастся, поэтому не будем отказываться от его использования. Но возьмем на заметку необходимость ограничения его возможностей для предотвращения несанкционированного доступа к ресурсам ОС. Необходимые мероприятия обсудим в следующем разделе.

Вариант 2. Сеанс Openbox

Поскольку, как мы только что выяснили, без оконного менеджера все равно не обойтись, возникла мысль: «А почему бы не использовать уже настроенный из коробки сеанс Openbox?» Давайте попробуем! Режим инфокиоска должен включаться только для пользователя kioskuser, поэтому выполним настройки сеанса Openbox именно для этой учетной записи. Зайдем в систему под пользователем kioskadmin, используя сеанс Openbox, – будем привыкать! (см. рис. 2)

Щелчком правой кнопки мыши на рабочем столе вызовем контекстное меню, выберем в нем пункт Terminal emulator (см. рис. 3) и получим права администратора:

```
$ sudo su -
```

Создадим локальные конфигурационные файлы на основе общесистемных:

```
# mkdir -p /home/kioskuser/.config/openbox
# cp /etc/xdg/openbox/* /home/kioskuser/.config/openbox
```

Назначение этих четырех файлов следующее [8]:

- environment** – сценарий установки среды окружения;
- autostart** – сценарий автоматического запуска приложений из оконного менеджера;
- rc.xml** – основные параметры оконного менеджера;
- menu.xml** – конфигурация главного меню оконного менеджера.

Среда окружения для нашей цели в настройке не нужна. В сценарии /home/kioskuser/.config/openbox/autostart напишем команду запуска браузера:

```
/usr/bin/chromium-browser
```

Для нормальной работы браузера нужно сделать kioskuser владельцем каталога /home/kioskuser/.config:

```
# chown kioskuser:kioskuser /home/kioskuser/.config
```

Чтобы оградить пользователя интернет-киоска от соблазнов, оставим в главном меню оконного менеджера единственную команду – Exit. Для этого файл /home/kioskuser/.config/openbox/menu.xml приведем к виду:

```
<?xml version="1.0" encoding="UTF-8"?>
<openbox_menu xmlns="http://openbox.org"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://openbox.org
    file:///usr/share/openbox/menu.xsd">
  <menu id="root-menu" label="Openbox 3">
    <item label="Exit">
      <action name="Exit" />
    </item>
  </menu>
</openbox_menu>
```

Завершим сеанс kioskadmin и зайдем в систему как kioskuser (тип сеанса снова выбираем Openbox). Если все было выполнено правильно, то на пустом рабочем столе должно появиться окно браузера Chromium. В противном случае нужно закрыть браузер и покинуть сеанс (в контекстном меню, вызываемом на рабочем столе, должен отображаться единственный пункт – Exit). В ходе настройки сеанса могут возникнуть ошибки, которые приведут к появлению системных сообщений (см. рис. 4).

Информация об ошибках записывается системой в файлы в каталоге /var/crash. Чтобы сообщения в дальнейшем не появлялись, эти файлы можно удалить.

Предположим, что в конце концов браузер у нас запустился. Но радоваться рано. Функции приложения, которые в обычной ситуации считаются неотъемлемыми, на компьютере общего пользования представляют угрозу безопасности системы. Прежде всего это доступ к файловой системе, который можно получить через (см. рис. 5):

- > ввод в адресной строке ссылок вида file:///etc/passwd;
- > сочетания клавиш для операций с файлами <Ctrl> + <O>, <Ctrl> + <S>;
- > контекстное меню, содержащее команду Save as...

Ограничение возможностей браузера Chromium

Пока еще у нас есть доступ к интерфейсу настройки браузера, самое время им воспользоваться для ограничения использования cookies и возможностей локального хранения данных браузером. Эти параметры зависят от сайтов и веб-приложений, с которыми предполагается работать. К сожалению, универсальных рекомендаций по этому вопросу дать нельзя.

Будем бороться с перечисленными выше проблемами в порядке их поступления. В браузере Chromium реализована опция командной строки «--app=<http://www.websaver.com>», приводящая к запуску его без строки ввода адреса [2]. Однако, если ссылка на странице запускает содержимое в новом окне, появляется полноценная копия браузера, с адресной строкой. К счастью, можно воспользоваться недокументированной опцией --kiosk [10]. Она действительно заставляет браузер выполняться в полноэкранном режиме. Обратите внимание на различия в синтаксисе опций --app и --kiosk: первая требует указания ссылки на веб-приложение после знака «равно», вторая опция просто является переключателем режима работы браузера. Эти опции являются взаимоисключающими, в командной строке должна использоваться лишь одна из них.

Всем хорош полноэкранный режим, но он решает лишь первую проблему. Странно, но разработчики браузера и создатели опции «--kiosk» убеждены, что доступ к файловой системе через другие каналы браузера должен пресекаться «внешними» средствами. Что ж, придется сейчас этим заняться.

Вспомним о конфигурационном файле оконного менеджера /home/kioskuser/.config/openbox/rc.xml. В нем, помимо прочих параметров, есть настройки «горячих клавиш». А графическая система устроена таким образом, что команды от органов управления сначала обрабатываются оконным менеджером, а потом передаются приложениям. Давайте попробуем назначить комбинация клавиш <Ctrl> + <O> и <Ctrl> + <S> какое-нибудь безопасное действие.

Например, переключение на следующее запущенное приложение, как это происходит при нажатии <Alt> + <Tab>. Для этого в секцию <keyboard> файла rc.xml нужно добавить такие блоки:

```
<keyboard>
...
<keybind key="C-O">
  <action name="NextWindow">
    <finalactions>
      </finalactions>
    </action>
  </keybind>
<keybind key="C-S">
  <action name="NextWindow">
    <finalactions>
      </finalactions>
    </action>
  </keybind>
<keybind key="C-O">
  <action name="NextWindow">
    <finalactions>
      </finalactions>
    </action>
  </keybind>
<keybind key="F12">
  <action name="NextWindow">
    <finalactions>
      </finalactions>
    </action>
  </keybind>
...
```

Последний блок перехватывает нажатие клавиши «F12», приводящее в нормальном режиме к выводу окон средств разработки Chromium. Практика показала, что действие

Рисунок 4. Системное сообщение об ошибке

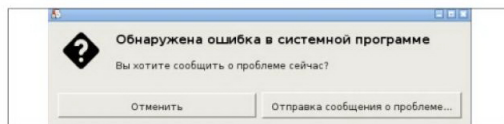
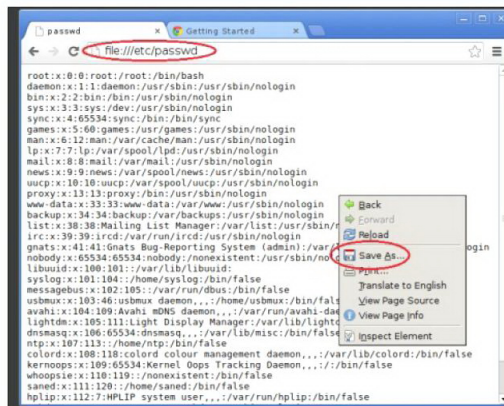


Рисунок 5. Лазейки к файловой системе, которые предоставляет браузер



«Следующее окно» является самым эффективным способом перехвата событий от средств ввода данных. Поскольку под управлением оконного менеджера запущен только браузер, это действие вполне безопасно.

По такой же схеме в секции <mouse> содержимое блоков:

```
<mousebind button="Right" action="Press">
...
</mousebind>
нужно заменить на
  <action name="NextWindow">
    <finalactions>
    </finalactions>
  </action>
```

Ограничим графический сеанс единственным виртуальным рабочим столом, для чего подправим секцию <desktops>:

```
<desktops>
...
  <number>1</number>
...
</desktops>
```

Ограничение возможностей вмешательства в работу системы

На данном этапе мы достигли того, что при входе в систему под профилем kioskuser с сеансом Openbox у нас стартует в монопольном режиме браузер Chromium, в котором отображается необходимый веб-ресурс. Возможности доступа к системе через браузер ограничены. Посмотрим, что еще может сделать потенциальный злоумышленник для того, чтобы использовать общественный компьютер не по назначению.

Вспоминаем, что с помощью комбинаций клавиш <Ctrl> + <Alt> + <F1> ... <Ctrl> + <Alt> + <F7> можно переключаться между виртуальными консолями, которые стартуют при загрузке Ubuntu и через которые можно получить сеанс доступа к системе. Проверим еще раз, работает ли у нас SSH-доступ, и принимаем решение о том, что наличие виртуальных консолей – неопозволительная роскошь.

Сконфигурируем PAM (Pluggable Authentication Modules, подключаемые модули авторизации) для запрета входа в систему с локальной консоли, оставим только разрешение на запуск графического сеанса пользователю kioskuser.

Для этого в файл /etc/security/access.conf добавим следующие строки:

```
+ : kioskuser : :0
- : ALL : LOCAL
```

Включим PAM-авторизацию, добавив в файлы /etc/pam.d/login и /etc/pam.d/lightdm строку:

```
account required pam_access.so
```

После перезагрузки убеждаемся в невозможности входа в систему с локальной консоли, пользователем kioskadmin подключаемся к компьютеру по протоколу SSH и смотрим журналы попыток доступа в файле /var/log/auth.log.

Для эстетического удовольствия запретим возможность переключения по виртуальным консолям, добавив в файл /etc/X11/xorg.conf секцию:

```
Section "ServerFlags"
    Option "DontVTSwitch" "true"
EndSection
```

После загрузки ОС будет появляться графическое окно авторизации пользователя, позволяющее осуществить вход в систему только под учетной записью kioskuser.

Автозапуск сеанса

Настроим менеджер дисплея lightdm на автоматический запуск созданного нами сеанса от имени kioskuser. Для этого создадим файл /etc/lightdm/lightdm.conf [9]:

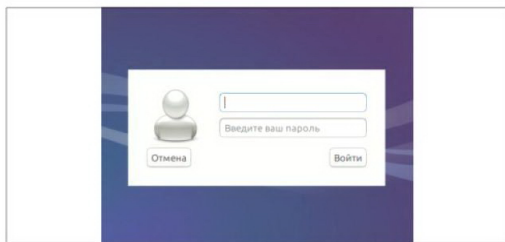
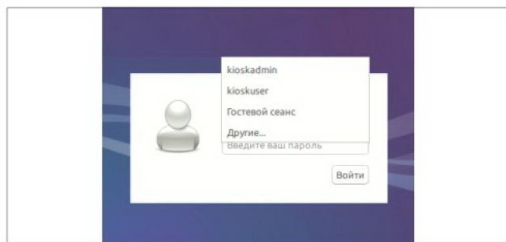
```
[SeatDefaults]
allow-guest = false
greeter-hide-users = true
autologin-user = kioskuser
autologin-user-timeout = 0
user-session = openbox
```

Первые две строки после названия секции – перестрочка. Они запрещают гостевой интерактивный сеанс и блокируют отображение списка учетных записей. Поэтому, если автозагрузка не произойдет, для входа в систему потребуется авторизоваться (см. рис. 6).

Автоматический вход в систему обеспечивают следующие две строки. Последняя, пятая, строка указывает название сеанса, который должен загружаться.

Самое время проверить, что у нас получилось. Для этого перезагрузим компьютер. Если все было проделано правильно, то после загрузки у нас на рабочем столе должно появиться окно Chromium.

Рисунок 6. Окно входа в систему: а) с раскрывающимся списком и гостевой учетной записью; б) без подсказок



Сколько еще не сделано...

Сеанс работы с браузером можно прервать с помощью комбинации клавиш <Alt> + <F4>. Если после этого вызвать контекстное меню и выбрать в нем пункт Exit, то система выведет приглашение к началу нового сеанса. Вероятно, это не лучший сценарий.

Давайте запретим завершение сеанса работы с браузером, для чего в файле /home/kioskuser/.config/openbox/autostart команду запуска браузера поместим в бесконечном цикле:

```
...
while true;
do /usr/bin/chromium-browser &
--kiosk http://www.webserver.com
sleep 5s;
done
```

Если в момент загрузки ОС удерживать клавишу <Shift>, то можно получить доступ в меню загрузчика GRUB (см. рис. 7).

Эту возможность тоже нужно заблокировать. Установим пароль администратора на специальные возможности меню Grub [12], для чего в файл /etc/grub.d/40_custom добавим три строки:

```
set superusers="kioskadmin"
password kioskadmin Pa$sw0rd
export superusers
```

В файле /etc/grub.d/10_linux добавим опцию --unrestricted в две строки, отвечающие за формирование пунктов меню загрузки:

```
echo "menuentry '${echo "$title" | grub_quote}'" \
$(CLASS) \${menuentry_id_option} \
'gnulinux-$version-$type-$boot_device_id' \
--unrestricted (" | sed "s/^/$submenu_indentation/"
echo "menuentry '${echo "$os" | grub_quote}'" \
$(CLASS) \${menuentry_id_option} \
'gnulinux-simple-$type-$boot_device_id' \
--unrestricted (" | sed "s/^/$submenu_indentation/"
```

Последнее необходимо сделать, чтобы пароль администратора не запрашивался при обыкновенной загрузке системы. После внесения изменений нужно дать команду на обновление параметров grub:

```
$ sudo update-grub
```

Осталось проверить меню BIOS/UEFI на предмет отключения дополнительных вариантов загрузки компьютера со сменных и подключаемых носителей, установить пароль на доступ к этому меню. Если к общественному компьютеру клавиатуру и мышь можно подключить по интерфейсу PS/2, то лучше использовать эту возможность, а USB-порты отключить. Системный блок желательно разместить в труднодоступном для пользователя месте.

...и сколько еще предстоит не сделать

Компьютер общего пользования готов к работе. Мы приложили много сил, чтобы сделать его достаточно защищенным. Но описанная схема настройки не является исчерпывающе полной и единственно верной.

Вместо браузера Chromium вполне можно использовать Firefox с надстройкой R-kiosk [11]. Разработчики надстройки уделили много внимания блокировке «опасных» возможностей браузера, в результате чего описанные в статье трюки с перехватом нажатий клавиш могут не потребоваться. Для серьезных применений можно вообще заменить браузер приложением собственной разработки.

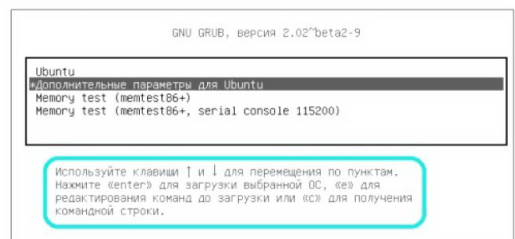
Использование свободных компонентов с открытым исходным кодом (ОС и браузера) благодаря наличию активного интернет-сообщества позволило нам решить поставленную задачу в сжатые сроки и с минимальными затратами. Полученное решение можно расширять, улучшать и адаптировать. Например, к компьютеру может потребоваться подключить локальный или сетевой принтер, для чего придется настроить систему печати CUPS и подправить настройки брендауэра.

Но стремление к совершенству бесконечно. В отличие от этой статьи. Удачных разработок! **BOF**

1. Отраслевой портал Kiosks.ru – <http://www.kiosks.ru/content/rus/rubr42/rubr-427.asp>.
2. Alan D Moore. Creating a kiosk with Linux and X11 – <http://www.alandmoore.com/blog/2011/11/05/creating-a-kiosk-with-linux-and-X11-2011-edition>.
3. Официальный сайт Ubuntu – <http://ubuntu.net>.
4. Образ диска Ubuntu 14.04 alternate i386 – <http://cdimage.ubuntu.com/ubuntu/releases/14.04/release/ubuntu-14.04-alternate-i386.iso>.
5. Зеркало официального репозитория Ubuntu – <http://mirror.vandex.ru/ubuntu/pool>.
6. Официальная документация Ubuntu – Iptables – <http://help.ubuntu.ru/wiki/iptables>.
7. Настройка Ubuntu в качестве веб-киоска – <http://www.instructables.com/id/Setting-Up-Ubuntu-as-a-Kiosk-Web-Appliance/?ALLSTEPS>.
8. Настройка оконного менеджера Openbox – <http://openbox.org/wiki/Configuration>.
9. Менеджер дисплея LightDM – <https://wiki.ubuntu.com/LightDM>.
10. Опции командной строки браузера Chromium – <http://peter.sh/experiments/chromium-command-line-switches>.
11. Надстройка браузера Firefox «R-kiosk» – <https://addons.mozilla.org/en-us/firefox/addon/r-kiosk>.
12. Официальная документация Ubuntu – Grub – <https://help.ubuntu.com/community/Grub2/Passwords>.

Ключевые слова: Ubuntu, Linux, инфокиоск, Iptables, Chromium.

Рисунок 7. Загрузчик GRUB предоставляет доступ к командной строке





Визитка

ДЕНИС СИЛАКОВ, кандидат ф.-м. н.,
старший системный архитектор ООО «НТЦ ИТ РОСА»,
denis.silakov@rosalab.ru

Установка и настройка Katello, Pulp и Candlepin

Знакомимся с новым поколением Red Hat Network

В статье о FOSDEM 2014 [1] отмечалось, что представители Red Hat на этой конференции активно рассказывали о линейке продуктов, задуманных как следующее поколение Red Hat Network — Katello, Pulp, Candlepin и Foreman. Познакомимся с первыми тремя приложениями

В рассматриваемой тройце:

Pulp — отвечает за управление репозиториями пакетов;

Candlepin — за управление подписками Red Hat;

Katello — предоставляет для этих двух инструментов удобный веб-интерфейс, заодно снабжая администраторов средствами управления пакетами и подписками для большого парка машин.

Развернув связку Katello + Pulp + Candlepin, мы получаем мощное средство управления пакетами на наших машинах, смещенное с сервисами подписок Red Hat.

Теоретически три рассматриваемых продукта независимы, но на практике Pulp и Candlepin должны стоять в одной системе с Katello. Неудивительно, что разработчики предусмотрели способ для прозрачной установки и настройки сразу всех трех продуктов. Именно этот процесс мы и рассмотрим.

Системные требования

Разворачивать серверную часть «Katello & со.» можно на RHEL с версии 6.4 и его производных, — таких, как CentOS и Scientific Linux. Мы в своих экспериментах использовали дистрибутив ROSA Enterprise Linux Server (RELS), также основанный на исходных кодах RHEL.

Что касается Fedora Linux, то Fedora 20 можно использовать относительно смело, Fedora 18 не поддерживается вовсе, а с Fedora 19 есть некоторые проблемы — обязательно почитайте инструкции на сайте Katello [2], если соберетесь ставить его на эту систему. Скорость развития проекта Fedora Linux такова, что в любой момент что-то может погнуться, а что-то, наоборот, сломаться, особенно в таком сложном продукте, каковым является стек из Katello, Pulp и Candlepin.

Из аппаратных архитектур поддерживается только x86_64. Минимальный объем ОЗУ — 2.5 Гб (при этом рекомендуется иметь минимум 4 Гб swap-раздела), для использования в реальных инфраструктурах рекомендуется минимум 8 Гб.

Для работы сетевых служб Katello необходимо открыть порты 80 (HTTP), 443 (HTTPS) и 5671 (AMPQ).

Наконец, несмотря на порывы избавиться от Java и перейти на Python, Katello все-таки использует Java и при этом поддерживает только OpenJDK, наличие которого является обязательным требованием к системе.

Установка

Рассмотрим самый простой случай — Katello ставится в систему, содержащую только базовую установку RHEL или RELS.

Если попытаться установить его на машину, куда уже устанавливалось какое-нибудь приложение из EPEL, использующее Ruby-on-Rails, то есть вероятность нарваться на поток ошибок. Также следует установить все обновления — например, к проблемам могут приводить устаревшие версии yum.

Для начала убедимся, что в системе команды 'hostname -s' и 'hostname -f' выдают имена, которые могут быть успешно разрешены (либо через DNS, либо через локальный файл /etc/hosts), а также успешно разрешается имя localhost. Проверить корректность настроек можно, выполнив в системе следующие три команды:

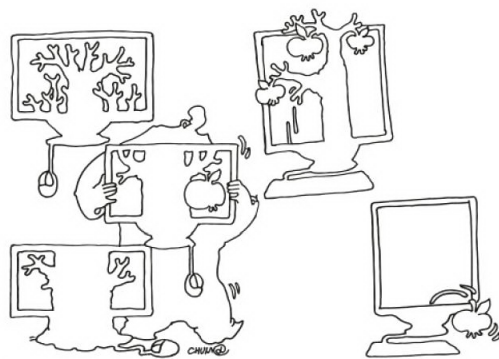
```
$ ping -c1 localhost
$ ping -c1 'hostname -s'
$ ping -c1 'hostname -f'
```

Каждая из них должна отработать успешно.

Теперь подключим репозитории Katello и Candlepin. Для этого установим в систему пакеты, добавляющие в настройку менеджера пакетов yum-информацию об этих репозиториях:

```
# rpm -Uvh http://fedorapeople.org/groups/katello/
releases/yum/1.4/RHEL/6Server/x86_64/
katello-repos-1.4.4-1.el6.noarch.rpm
# rpm -Uvh http://dl.fedoraproject.org/pub/epel/6/
x86_64/epel-release-6-8.noarch.rpm
```

Также нам понадобится репозиторий EPEL — в этом случае нам придется скачать файл с настройками репозитория и поместить его в директорию с настройками yum:



Red Hat Network — мощное средство управления пакетами на машинах, совмещенное с сервисами подписок

```
# wget -O /etc/yum.repos.d/epel-rhsm.repo -J
http://repos.fedorapeople.org/repos/candlepin/ -J
subscription-manager/epel-subscription-manager.repo
```

Эти рекомендации работают для RHEL/CentOS и им подобных.

Для Fedora репозитории могут немного отличаться, а в случае Red Hat разработчики вместо добавления репозитория Candlepin с [fedorapeople.org](http://repos.fedorapeople.org) рекомендуют просто подписать систему на соответствующие репозитории RHEL:

```
# yum-config-manager --enable rhel-6-server-optional-rpms
```

или

```
# rhn-channel --add --channel=rhel-x86_64-server-optional-6
```

Наконец, нехитрым действием установим все, что нужно для Katello:

```
# yum install -y katello-all
```

Немного терпения, и все необходимые пакеты будут установлены в вашей системе.

Базовая настройка сервера

После установки Katello необходимо сконфигурировать. Делается это полностью автоматически с помощью утилиты `katello-configure`, которой можно передать ряд базовых параметров:

```
# katello-configure --user-name=admin --user-pass=123456 -J
--user-email=admin@rosalab.ru --org-name=ROSA
```

Здесь же мы указали адрес электронной почты, на который будут отсылаться все уведомления от Katello, а заодно создали организацию ROSA, к которой будем прикреплять все подписки («организация» в Katello — это то же самое,

что и «организация» в Spacewalk, и при необходимости вы можете создавать несколько организаций и определять для каждой из них свой набор подписок).

Katello-configure берет на себя практически всю работу по развертыванию и настройке всех компонентов сервера, включая веб-службы и базу данных. Остается только налить себе кофе и немного подождать, впрочем, на быстром сервере вы и кофе допить не успеете, как веб-интерфейс Katello станет доступен по адресу https://_имя_хоста/_katello.

Пора зайти в него под пользователем `admin` с паролем `123456` и перейти непосредственно к наполнению содержимым нашей инфраструктуры.

Репозитории и подписки Red Hat

Основное предназначение нашей связки Katello + Pulp + Candlepin — это управление продуктами и подписками на них. Напомню, что в терминологии RHN продукт — это набор репозиториями пакетов, с которым можно работать как с атомарной сущностью. Подключение системы к этому набору репозиториями называется подпиской системы на продукт.

Подписки используются не только для удобства работы с множеством репозиториями, но и для распространения обновлений на платной основе (в частности, обновлений для дистрибутивов Red Hat Enterprise Linux, которые доступны публично только в виде исходных кодов, но не в виде собранных пакетов).

Поэтому каждая подписка характеризуется датами ее начала и окончания (в течение периода действия подписки пользователь сможет получать обновления из репозиториями продукта), а также максимальным количеством машин, на которое она распространяется (то есть сколько машин можно подключить к репозиториями продукта по этой подписке).

Для простоты давайте добавим внешние репозитории, на основе которых мы будем формировать продукты и раздавать их клиентам Katello в нашей сети. Этот процесс производится в меню Content → Repositories → Products.

Здесь можно нажать на кнопку New Product, создать новый продукт, а затем присоединять к нему репозитории по одному.

Однако мне больше понравился альтернативный способ: нажать кнопку Repo Discovery и предоставить ссылку на репозиторий либо на страницу, содержащую сразу несколько репозиторий. Katello просканирует указанную ссылку, выведет список найденных репозиторий и предложит вам выбрать те, которые вам интересны.

Katello, Pulp и Candlepin с точки зрения функционала и качества оставляют впечатление готовых к использованию инструментов

Например, указав в качестве исходной ссылки http://mirror.rosalab.ru/rosa/server/6.5/repository/x86_64, я получил все 64-битные репозитории ROSA Server 6.5. После этого можно подключить выбранные репозитории к имеющемуся продукту либо создать на их основе новый (см. рис. 1).

Настроить расписание синхронизации с внешними репозиториями и отслеживать статус синхронизации можно в меню Content → Sync Management. Обратите ваше внимание, что процесс синхронизации, а особенно первоначального импорта всех внешних репозиторий, отнюдь не быстрый.

Дело в том, что используемый для управления репозиториями Pulp не просто скачивает пакеты в свой каталог, но анализирует метаданные репозитория и раскладывает пакеты в своих внутренних директориях согласно их имени и версии.

Только после этого формируется собственно репозиторий на стороне Pulp, причем делается это просто созданием в папке репозитория ссылок на файлы с пакетами, расположенными во внутренних директориях сервера. За счет

этого можно получить серьезную экономию дискового пространства в случае, если вы подключаете множество репозиторий, имеющих схожие наборы пакетов.

Разработчики Katello не забыли о ситуации, когда Katello разворачивается на сервере, не имеющем прямого доступа в Интернет. Для таких случаев создана утилита `katello-disconnected`, о тонкостях использования которой можно прочитать на вики проекта [3].

Что касается подписок, то вы можете импортировать манифесты имеющихся у вас подписок Red Hat в меню Content → Subscriptions → Red Hat Subscriptions. Более того, раздачу самих подписок через сервер Katello – для этого необходимо перейти в меню Content → Subscriptions → Subscription Manager Applications, где можно добавлять дистрибьюторов (нажимая кнопку New Distributor), делегируя им часть доступных подписок.

Подключение клиентов

Подключение машин-клиентов к серверу Katello производится непосредственно с этих машин. Здесь я опишу действия, которые необходимо проделать на каждой машине, но делать их вручную совсем не обязательно – в документации Katello можно найти примеры скриптов для автоматического подключения клиентских машин к серверу.

Для регистрации клиента нам в первую очередь понадобится пакет `subscription-manager` – в RHEL/RELS этот пакет установлен по умолчанию, в Fedora его надо поставить вручную, но рекомендуется всегда обновляться до последней версии.

```
# yum install subscription-manager
```

Также необходимо установить пакет `katello-agent` – именно с помощью этого агента будет производиться удаленное управление пакетами машины, он же будет предоставлять различные сведения о состоянии ОС и выполнять другие вспомогательные функции.

```
# yum install katello-agent
```

Рисунок 1. Katello Repo Discovery

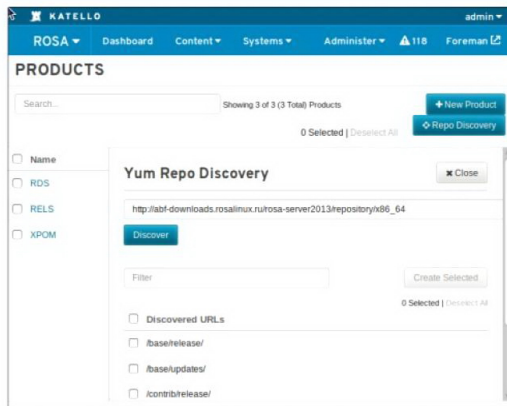
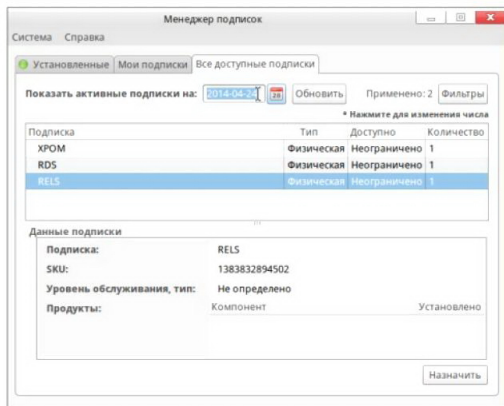


Рисунок 2. Подключаем клиента к Katello через GUI



Перед подключением клиента необходимо будет скачать и установить с сервера подписочный пакет с сертификатом и настройками для subscription-manager, например, сделав это так:

```
# rpm -Uvh https://имя_хоста_katello/pub/ ./
candlepin-cert-consumer-latest.noarch.rpm
```

Этот пакет содержит сертификат сервера и скрипт, который модифицирует конфигурационный файл /etc/rhsm/rhsm.conf для работы с вашим сервером Katello (вместо RedHat Networks, на который subscription manager настроен по умолчанию в RHEL).

Установить сертификат и внести необходимые в него изменения в файл rhsm.conf можно и вручную, без установки – соответствующие инструкции можно найти в документации Katello.

И, наконец, регистрируем машину в Katello:

```
# subscription-manager register --org ROSA
```

В опции --org надо указать существующее в Katello имя организации. Дополнительно с помощью опции --activationkey можно передать ключ, к которому уже привязаны необходимые подписки.

Можно воспользоваться и графическим интерфейсом, запускаемым с помощью subscription-manager-gui (для этого потребуются установить одноименный пакет, если он у вас еще не установлен) (см. рис. 2).

Далее вас попросят указать имя и пароль пользователя Katello, от имени которого будет произведена регистрация; эти параметры можно передать в командной строке с помощью опций -i и -p соответственно.

Если все прошло успешно, subscription-manager вернет вам uuid, присвоенный клиентской машине. Теперь можно посмотреть список доступных подписок:

```
# subscription-manager list --available
```

И подписаться на одну из них можно командой subscription-manager subscribe либо используя графический интерфейс. Последний удобнее для использования человеком, но при автоматическом развертывании без консольного клиента не обойтись.

С помощью subscription-manager на клиентской машине можно заодно и привязаться к конкретной версии дистрибутива (атрибут releasever в терминах yum) – например, указать, что для данного клиента необходимо ставить пакеты от RHEL/RELS 6., можно с помощью команды

```
# subscription-manager release --set=6.1
```

Впрочем, администраторы, которые хоть раз имели дело с необходимостью указать значение releasever для yum, заметят, что того же эффекта можно добиться, просто прописав нужное значение в файл /etc/yum/vars/releasever.

В процессе подписки в систему будут добавлены соответствующие репозитории, которые можно использовать

Рисунок 3. Через Katello можно получить различные данные о клиентских машинах

The screenshot shows the Katello web interface. The top navigation bar includes 'ROSA', 'Dashboard', 'Content', and 'Systems'. The 'Systems' section is active, showing a list of systems on the left and a detailed view of a selected system on the right. The selected system is 'rosa-desktop-fresh2'. The details view is divided into several tabs: 'Details', 'Subscriptions', 'Events', 'Packages', and 'Errata'. The 'Details' tab is currently selected, showing 'Basic Information' and 'System Content'. The 'Basic Information' section includes fields for Name, UUID, Description, and Type. The 'System Content' section includes fields for Release Version, Content View, and Environment. The 'Subscriptions' section shows the Subscription Status, Auto-Attach, Service Level, and Activation Key(s). The 'System Properties' section shows OS, Release, System Groups, Arch, Number of CPUs, Sockets, and Cores per Socket. The 'System Status' section shows the Registered status and Checkin time. The 'Networking' section shows Hostname, IPv4 Address, IPv6 Address, and Interfaces.

для установки и удаления пакетов. Клиентская машина теперь должна быть видна в веб-интерфейсе Katello. Рассмотрим вкратце, что полезного можно сделать с помощью этого интерфейса в настоящее время.

Работа с клиентскими машинами

Список всех клиентов, подключенных к Katello, доступен в меню Systems → All. Для просмотра информации и управления им просто кликните на соответствующее имя в списке. На появившейся вкладке увидите как базовые сведения о подключенном клиенте (версию ОС, имя машины, активированные подписки), так и детальную информацию о некоторых аппаратных характеристиках машины (ЦПУ, ДМЛ, оперативной памяти, используемой виртуализации) и ее настройках сети в секции Advanced Information (см. рис. 3).

Вкладка Subscriptions служит для управления подписками клиента, на вкладке Events можно просматривать некоторые события, о которых «докладывает» агент Katello, на вкладке Errata управлять списками Errata, и, наконец, наиболее интересный аспект – управление пакетами на вкладке Packages. Для каждой клиентской машины на этой вкладке вы можете посмотреть список установленных пакетов, а также установить, обновить или удалить конкретные пакеты или группы пакетов (группы – в терминах yum) (см. рис. 4).

Работать с каждой конкретной клиентской машиной в большой инфраструктуре – занятие малоэффективное, и Katello предлагает возможность осуществлять управление пакетами сразу на множестве машин. Для этого на странице Systems необходимо нажать кнопку Bulk Actions, отметить в списке слева компьютеры, для которых будет выполняться операция, и, собственно, выбрать операцию – можно объединить выбранные системы в группу, установить, удалить или обновить на них пакет или группу пакетов, а также активировать все доступные для этих систем подписки (см. рис. 5).

Объединение систем в группы полезно в случае необходимости производить какие-то манипуляции с одним и тем же набором машин. Для работы с группами предусмотрен другой пункт меню Systems → System Groups. Принцип работы с группами такой же, как и с отдельными пакетами при выполнении Bulk Action, – просто выберите группу из списка и в правой части окна сможете манипулировать пакетами сразу всех членов группы.

В целом Katello и сопутствующие инструменты с точки зрения функционала и качества оставляют впечатление готовых к использованию, во всяком случае, каких-то критических ошибок во время наших экспериментов мы не встретили. Эксперименты проводились на стенде из одного сервера и десятка виртуальных машин в роли клиентов – конечно, это не предприятие с сотнями компьютеров, но принципы и алгоритмы работы одни и те же.

Однако следует помнить, что новый стек приложений для формирования Red Hat Networks сейчас находится в активной разработке, и пока официальной миграции RHN на новые продукты не случилось, использовать их можно исключительно на свой страх и риск. Следствием активно ведущейся разработки является и недостаточная документированность многих аспектов работы – например, руководство пользователя Katello [4] пока что описывает только основные функции, да и то не все.

С другой стороны, веб-интерфейс Katello интуитивно понятен (особенно если вы уже имели дело с Satellite или Spacewalk), и базовых сведений из руководства пользователя должно хватить для его настройки и администрирования.

В заключение отметим, что к связке Katello, Pulp и Candlepin можно добавить еще и Foreman – инструмент для управления полным жизненным циклом пула серверов, включая развертывание ОС и управление конфигурацией. Однако Foreman является полностью независимым продуктом, его интеграция с Katello сейчас минимальна, и он достаточно сложен и интересен, чтобы посвятить ему отдельную статью. **БОЕ**

1. Силаков Д. FOSDEM 2014. Впечатления о крупнейшей Open Source-конференции в Европе. // «Системный администратор», №3, 2014 г. – С. 4-7 (<http://samag.ru/archive/article/2633>).
2. Katello Installation Instructions – <https://fedorahosted.org/katello/wiki/Install>.
3. Katello Disconnected – развертываем сервер Katello в сети без доступа к Интернету – <https://fedorahosted.org/katello/wiki/GuideKatelloDisconnected>.
4. Katello User Guide – <https://fedorahosted.org/katello/wiki/UserGuide>.

Ключевые слова: Red Hat, Katello, управление, Linux.

Рисунок 4. Оперирем с пакетами на клиентской машине

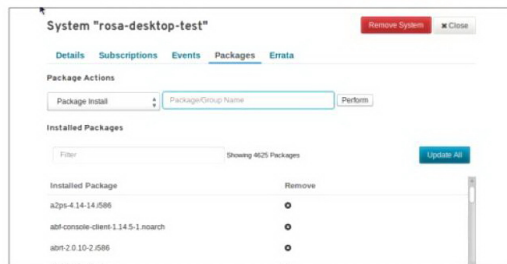
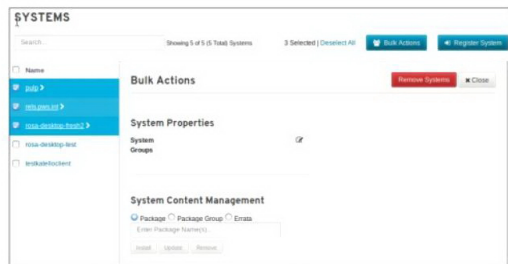


Рисунок 5. Bulk Actions позволяют обновлять пакеты сразу на множестве машин



Визитка

СЕРГЕЙ ЖИТИНСКИЙ,
генеральный директор Git in Sky (ООО «Жить в небе»),
sergey@gitirinsky.com



Визитка

АЛЕКСАНДР ЧИСТЯКОВ,
главный инженер Git in Sky (ООО «Жить в небе»),
alex@gitirinsky.com



Системы управления конфигурацией Чем отличаются Chef, SaltStack, Puppet, CFEngine и Ansible?

В статье рассматриваются четыре важнейшие группы свойств систем управления конфигурацией, по которым делается сравнение популярных СМ-инструментов

Компьютеры, которые работают в качестве серверов, должны обладать в первую очередь повышенной надежностью и бесперебойностью в работе. Для этого, кроме прочего, им нужно обеспечить очень хорошую управляемость. Это достигается многими технологиями и, в том числе, применением программных систем управления конфигурациями. По-английски эти системы называются Configuration Management (CM).

На рис. 1 изображена типичная организация компонентов программного СМ-инструмента в общем случае. Operator – это вы – системный инженер, который эксплуатирует систему. В Repository хранятся вводные данные – спецификации, которые описывают инфраструктуру или сервер. Спецификация переводится в profile на Translation Agent. Эти данные передаются (или забираются) на управляемые узлы (managed devices) и превращаются там в набор исполняемых команд при помощи компонента внедрения (deployment agent).

СМ-системы позволяют настраивать сервера не руками, а программным способом. Они автоматизируют труд системных администраторов, превращая их в системных инженеров. Серверы при этом настраиваются быстро и безошибочно (если, конечно, сами управляющие скрипты-рецепты-плейбуки были написаны без ошибок). Кроме того, система управления конфигурацией позволяет следить за изменениями состояния сервера и поддерживать на нем стабильную эталонную конфигурацию.

Решение подобной задачи необходимо, как правило, на довольно крупных системах, так как позволяет экономично эксплуатировать большой парк однотипных серверов. Однако теперь все чаще можно встретить использование СМ на небольших проектах, изначально закладывающих в архитектуру несколько ролей серверов и стандартных окружений (разработка-тест-продакшн).

Управление конфигурацией через код, хранимый в репозитории, существенно повышает управляемость процес-

Рисунок 1. Общая схема работы СМ-системы



сом настройки и администрирования ИТ-инфраструктуры. Эта парадигма приобрела даже свою собственную аббревиатуру – IaC – Infrastructure as a Code.

Поскольку интерес к СМ-продуктам неуклонно растет, в данной статье мы попытались проанализировать и сравнить самые популярные из них.

История возникновения популярных СМ-инструментов

Первая система управления конфигурацией, появившаяся на рынке, была CFEngine. Основатель и бессменный руководитель проекта CFEngine – норвежский ученый Марк Бургес, работающий в университете города Осло.

Первая версия продукта вышла в 1993 году. По понятным причинам основными параметрами системы должны были стать легковесность и нетребовательность к ресурсам, особенно к сетевым. Действительно, особенности архитектуры позволяли настраиваемым компьютерам преспокойно жить даже в отсутствии связи с управляющей машиной.

Почти одновременно, в 1994 году, выходит LCFG – СМ-система на Perl, разработанная в Эдинбургском университете. В отличие от CFEngine, система использовала централизованную архитектуру с обработкой запросов узлов на центральном сервере.

В 1998 году появляется еще несколько систем – ISConf, PIKT, STAF. Честно говоря, мы не исследовали их подробно, важно, что на этом рынке уже была конкуренция.

В 2002 году выходит CFEngine2 – версия, в которой вводятся понятия *convergence to desired state* и *self-healing*. Эти парадигмы позволяли управляемым компьютерам самостоятельно обнаруживать отклонения от желаемого состояния и возвращаться к нему путем выполнения определенных команд или операций.

В 2000-е годы появляется еще несколько разработок, из которых самой популярной становится BCfg2 на Python. В этой системе не было изначально модульности и она не позволяла подняться выше уровня абстракции «Configuration files» (см. ниже). Но система была написана на Python и поэтому привлекла к себе внимание сообщества «питонщиков».

В 2005 году один из продвинутых пользователей CFEngine2 Luke Kanies решает сделать свой продукт, в связи с тем, что он был недоволен тем, как Марк Бургес управлял развитием CFEngine.

Действительно, ученые, при всем к ним уважении, редко бывают хорошими управленцами или бизнесменами. Так родился Puppet, отличительными особенностями которого стал движок моделей на основе графов состояний, простой и понятный DSL-язык описаний на основе Ruby

и изначальная ориентированность на сообщество разработчиков. Все эти инновации были востребованы рынком, поэтому Puppet стал активно внедряться в промышленную эксплуатацию.

В 2008 году Марк Бургес и его проект CFEngine3 ответили Puppet более удобным языком описаний и новой стройной концепцией, которая называлась *Theory of Promises* – теория обещаний. В одноименной книге Бургес попытался описать «обещания» как технический объект, а не юридически-моральный, как было принято до него. Концепция была внедрена в продукт и теперь, если вы хотите настроить сервер при помощи CFEngine3, вы должны разобраться, что такое «обещание» и кто кому их дает среди серверов вашего проекта.

В 2009 году несколько активных членов сообщества Puppet во главе с Adam Jacobs в свою очередь решили отпочковаться и создать свой продукт, который они назвали Chef. В отличие от Puppet, и его якобы «недетерминированности», в Chef все операции идут в том порядке, в котором их написал программист. Кроме того, язык описаний (DSL) в Chef – это просто Ruby, что позволило привлечь разработчиков на этом популярном языке к созданию первичной базы скриптов (в терминологии Chef – рецептов).

В 2011 разработчики суперскоростного пакета обмена сообщениями ZeroMQ создают свою СМ-систему SaltStack, преимуществом которой является именно скоростная коммуникация между узлами сети.

В 2012 стартовал проект Ansible – система управления конфигурациями, с возможностью решения ad-hoc задач, без софтверного агента на управляемом узле. Работает через SSH, на управляемом узле требуется только Python версии 2.4+. Как следствие эта программа не позволяет управлять зависимостями между узлами инфраструктуры.

Как видим, пакетов достаточно много, познакомиться с ними можно в Википедии [1]. Причем это сравнение не содержит проприетарные инструменты от компаний IBM, HP, Microsoft, Dell и других серьезных производителей решений.

Мы выбрали для анализа только самые известные из Open Source решений, которые используются на реальных проектах и, что немаловажно, используются реальными клиентами нашей компании Git in Sky, то есть мы имели опыт работы с ними в реальных задачах.

Итак, эти инструменты:

- > CFEngine3 (Community Edition v3.5.0);
- > Puppet (v2.7.*);
- > Opscode Chef (v11.6.0);
- > SaltStack (v2014.1.0);
- > Ansible (v1.5).

Таблица 1. Базовые свойства языка описания

СМ-инструмент	Декларативный/императивный DSL	DSL использует язык/стандарт	Наличие Web-интерфейса
CFEngine 3	Декларативный	Свой	Да, в платной версии
Puppet	Декларативный/Императивные элементы	Свой/Ruby	Да
Chef	Императивный/ Декларативные элементы	Ruby (в чистом виде)	Да, только управление рецептами/кукбуками
Salt	Декларативный/императивные элементы	YAML/Python	Да, пока сырая версия
Ansible	Декларативный/императивные элементы	YAML	Да, только в платной версии Ansible Tower

Методология сравнения

В своем исследовании мы опирались на методологию [2], разработанную в 2010 году в бельгийском университете Katholieke Universiteit Leuven.

Для сравнения мы выделили следующие группы свойств продуктов:

1) Свойства системы ввода данных об управляемой системе серверов.

В этом разделе мы сравниваем то, как разные продукты предписывают вводить информацию о настройках серверов. Как правило, используется некий язык настройки, на котором описываются конфигурации системы и/или отдельных узлов.

1.1) **Базовые свойства языка описания (DSL).** Системным отличием таких языков является либо декларативное, либо императивное описание. Кроме того, в этой системе мы оцениваем наличие возможности управления через веб-страницу. (Таблица 1)

Декларативный стиль больше подходит для описания состояний серверов, в то время как императивный позволяет осуществлять полный контроль за процессом внедрения изменений. Веб-интерфейс редко бывает нужен профессиональным системным администраторам или инженерам, но его наличие, тем не менее, позволяет специалистам облегчить свой труд, в частности, по построению отчетов и наглядному отображению состояний инфраструктуры.

1.2) **Механизмы абстракции** – это наличие возможностей в языке описания задавать те или иные уровни абстракции от конкретных реализаций. Можно выделить шесть таких уровней:

а. Управление требованиями SLA к системе (например: «сконфигурировать достаточно много почтовых серверов, чтобы обеспечить время SMTP-отклика N ms»).

б. Управление требованиями к ролям (классам) инстансов (например: «сконфигурировать N подходящих машин в качестве почтовых серверов»).

с. Управление требованиями к инстансам независимо от их конкретной реализации (например: «сконфигурировать машины X,Y,Z как почтовые серверы»). При этом инстансы могут разбиваться на ряд параметров. (например, параметрами веб-сервера будут – порт, который он слушает; поддерживаемые скриптовые языки; возможность проксирования и т.д.).

д. Управление требованиями к конкретным реализациям инстансов (например: «Создать/заменить следующие строчки в sendmail.cf на машинах X,Y,Z»).

е. Управление конфигурационными файлами (например: «положить конфигурационный файл sendmail.cf на сервер X»).

ф. Управление двоичными образами конфигураций («задать образ виртуальной машины на сервер X», см. таблицу 2).

1.3) Механизмы поддержки модульности, возможности по сокращению повторяющихся задач.

Для сокращения трудозатрат и увеличения повторяемости использования кода необходим тот или иной уровень модульности, предусмотренный в системе. В общем случае вопрос повторяемости использования кода сводится к описаниям групп узлов по разным параметрам, чтобы применять к ним один и тот же код. В более продвинутых слу-

чаях эти группы определяются динамически по свойствам инстансов, в зависимости от ответа на тот или иной запрос или в зависимости от тех или иных настроек окружения. Дополнительным свойством механизма модульности является возможность создавать модули третьими сторонами (не разработчиком инструмента, сообществом).

Возможные механизмы:

- > Статически описанные группы (Static).
- > Группы сформированные по запросу (Query based).
- > Иерархические группы (Hierarchical).

1.4) Механизмы моделирования отношений между настраиваемыми сущностями.

Один из главнейших источников ошибок и простоев в инфраструктуре – неправильные конфигурации из-за человеческих ошибок. Типичный случай – несогласующаяся или противоречивая конфигурация. Например, если вы переносите зону ДНС на другой NS сервер и забыли изменить его IP-адреса. Явно заданные модели отношений, которые присутствуют в сетевых конфигурациях позволяют сохранить конфигурационную модель согласованной и непротиворечивой. Например, явно заданное отношение между настройками DNS клиента и адресом DNS-сервера позволяет при его переносе на другой адрес автоматически сделать изменения на всех клиентах. Моделирование отношений позволяет существенно сократить риск ошибок при управлении конфигурациями.

Таблица 2. Механизмы абстракции

CM-инструмент	Возможные уровни абстракции
CFEngine 3	C, D, E
Puppet	C, D, E
Chef	C, D, E
SaltStack	C, D, E
Ansible	C, D, E

Таблица 3. Механизмы поддержки модульности, возможности по сокращению повторяющихся задач

CM-инструмент	Механизмы поддержки модульности	Возможность создания модулей 3 сторонами
CFEngine 3	A,B,C	Да
Puppet	A,B,C	Да
Chef	A,B	Да
SaltStack	A,B	Да
Ansible	A	Да

Таблица 4. Механизмы наложения ограничений на отношениях

CM-инструмент	Детальность	Ариность	Ограничения
CFEngine 3	A, B, C	A, B, C	B
Puppet	A, B, C	B, C	A
Chef	A, B, C	B, C	A
SaltStack	A, B, C	B, C	A
Ansible	Не поддерживает	Не поддерживает	Не поддерживает

Существует несколько механизмов моделирования отношений:

Механизм детальности отношений моделирует отношения между конфигурационными узлами (инстансами) и их параметрами. В разделе 1.2 мы говорили о конфигурировании инстансов (класс C) и о том, что они могут быть разбиты на параметры. Такие отношения могут быть трех типов:

A. Инстанс-инстанс. Примером такого отношения является приведенный выше пример зависимости между инстансом сервера DNS и его клиентов.

B. Параметр-параметр. Примером такой зависимости является, например, CNAME-запись, которая требует обязательного наличия соответствующей A-записи в DNS.

C. Параметр-инстанс. Примером такой зависимости может быть зависимость mail-сервера от наличия MX-записи в DNS.

Механизм арности (валентности) отношений моделирует отношения между тем или иным количеством сущностей. Эти отношения могут быть следующих типов:

A. One-to-one.

B. One-to-many.

C. Many-to-many.

Механизмы наложения ограничений на отношениях позволяют ввести в описание отношений между сущностями необходимые ограничения, они могут быть двух типов:

A. Ограничение с проверкой условия. Например: «Сервер может обслуживать не более 100 клиентов» или «Клиенты могут использовать DNS-серверы, доступные в их подсети».

B. «Мягкое» ограничение. Такое ограничение оставляет больше свободы для CM-инструмента, с возможностью вы-

бора одного решения из ряда, например «сконфигурировать N машин и одна из них должна быть mailserver» (см. таблицу 4).

2) **Свойства, связанные с внедрением описанных конфигураций.**

2.1) **Масштабируемость.** При росте количества управляемых узлов, возникает большая нагрузка на конфигурационные сервера, особенно в режиме «pull». Кроме того, чем больше инфраструктура, тем она более разнородна, что заставляет использовать более высокие уровни абстракции, а это в свою очередь приводит к дополнительной нагрузке при переводе абстрактных правил в конкретные для каждого из узлов.

То, как CM-инструмент решает эти проблемы, определяет его масштабируемость.

2.2) **Архитектура процесса внедрения конфигураций.** Здесь CM-продукты можно различать по соотношению количества translation agents (см. рис. 1) и количества управляемых узлов (managed nodes):

A. Централизованная (один центральный сервер)

B. Слабо распределенная – может быть несколько серверов (например, реплик) на много управляемых узлов.

C. Сильно распределенная – для каждого узла свой translation agent.

Кроме того CM-инструменты нужно различать по **способу доставки** Push/Pull. Push подразумевает «проталкивание» изменений от сервера, а Pull – запрос управляемым узлом сервера о возможных изменениях.

2.3) CM-инструменты также необходимо различать по **списку поддерживаемых платформ** для управления. Не все системы поддерживают все многообразие серверных платформ. Кроме того, некоторые инструменты под-

Таблица 5. Список поддерживаемых платформ

CM-инструмент	Масштабируемость	Deployment architecture	Способ доставки	Platforms
CFEngine 3	> 10000 узлов	C	Pull, Push	*BSD, AIX, HP-UX, Linux, Mac OS X, Solaris, Windows
Puppet	Есть инсталляции 100k узлов	A,B	Push, Pull	*BSD, AIX, HP-UX, Linux, Mac OS X, Solaris, Windows
Chef	>10000, заявлено до 50000	A,B	Pull, Push (Push только в модуле к Enterprise Chef)	*BSD, Linux, Mac OS X, Solaris, Windows
SaltStack	Нет информации	A,B	Pull	*BSD, AIX, HP-UX, Linux, Mac OS X, Solaris, Windows
Ansible	> 100	A	Push	*BSD, AIX, HP-UX, Linux, Mac OS X, Solaris

Таблица 6. Механизмы контроля доступа

CM-инструмент	Простота	Тесты	Мониторинг	Версионность	Документирование	Окружения	Права
CFEngine 3	Сложно	Dry run	Встроен	Внешний репозит	Структурированная	Runtime	Path-based
Puppet	Средне	Dry run, staging/testing integration	Интеграция с внешним	Внешний репозит	Свободная и структурированная	Runtime, database	Path-based
Chef	Сложно	Dry run, staging/testing integration	Интеграция с внешним	Встроенный и внешний репозит	Свободная и структурированная	Runtime, database	Path-based
SaltStack	Средне	Dry run	Встроен с 0.15.0	Внешний репозит	Свободная	Runtime, database	Встроенный ACL
Ansible	Просто	Dry run	Интеграция с внешним	Внешний репозит.	Свободная	Runtime	Нет

держивают возможность настройки сетевых устройств, принтеров и других устройств. В эту же группу свойств можно отнести возможность взаимодействия с другими CM-инструментами. (таблица 5)

3) Большая группа свойств CM-инструментов связана с вопросами **управления входными спецификациями**. Это включает в себя:

3.1) **Usability** или удобство использования, которое можно разбить на 3 аспекта:

3.1.1) **Простота языка описаний**. Язык описаний должен быть понятен и достаточно быстро осваиваться системными администраторами, у которых необязательно есть высшее образование. Некоторые инструменты подразумевают достаточно высокий опыт разработки, который мы назовем DevOps Skills.

3.1.2) **Поддержка тестирования спецификаций**. Инструменты должны (или могут) предоставлять возможности по тестированию измененных или новых спецификаций путем «dry run» прогона всех команд без реального внесения изменений или возможности построения тестовой виртуальной среды.

3.1.3) **Поддержка мониторинга управляемой инфраструктуры** – какие возможности предлагает инструмент по организации мониторинга созданной при его помощи инфраструктуры.

3.2) **Поддержка версионности спецификаций** – обычно используются репозитории и программы контроля версий. Все ли инструменты поддерживают это?

3.3) **Поддержка документирования спецификаций** – возможность генерации документации из кода.

3.4) **Поддержка организационных политик** в процессе внедрения изменений – возможность **создания окружений** (staging, testing) и проведения необходимых стадий отладки продукта.

3.5) **Механизмы контроля доступа** – поскольку каждый сотрудник, имеющий доступ к системе составления спецификаций и их сохранению, фактически имеет доступ ко всей управляемой инфраструктуре, необходимы эффективные механизмы разграничения прав. Это особенно актуально на больших инфраструктурах, когда десятки системных администраторов с разными правами должны иметь доступ к одной и той же системе управления спецификациями CM-инструмента. (см. таблицу 6)

4) Свойства, связанные с поддержкой продукта.

4.1) Насколько просто освоить данный продукт? Здесь мы оцениваем все продукты по 5-балльной шкале исходя из собственного опыта освоения. 1 – трудно, 5 – легко.

4.2) Насколько просто внедрить его в реальную систему? Наша субъективная 5-балльная оценка. 1 – трудно, 5 – легко.

4.3) Наличие документации. Качество документации, наличие tutorиалов, обучения.

4.4) Наличие коммерческой поддержки – обучение, возможность настройки решений под Заказчика

4.5) Наличие сообщества пользователей – по 5-балльной шкале мы оценили размер и активность сообщества

4.6) Степень зрелости продукта – оценка степени развития, не появляются ли с каждой новой версией достаточно серьезные изменения и функции. Степень поддержки предыдущих версий (см. таблицу 7)

В компании Git in Sky, внедрили более 6 проектов на Chef, 2 проекта на Puppet, 4 проекта на SaltStack и управляем на Ansible своей собственной инфраструктурой. Кроме всего перечисленного в статье мы можем сказать о том, что несмотря на то, что все инструменты могут управлять любой инфраструктурой, Chef из первичной установки сразу хорошо работает с Debian/Ubuntu, а под CentOS/RHEL требует больше отладки. Puppet прекрасно работает с CentOS/RHEL, а SaltStack живет достаточно хорошо на все Linux.

Ansible, в силу своей ad-hoc природы прекрасно работает через SSH с любой ОС, но на SmartOS он собирает слишком мало фактов. Впрочем, работа со SmartOS достаточно слабо отражена во всех продуктах, т.к. эта ОС мало распространена.

Можем сделать вывод, что для управления сложными системами нужно использовать первые из трех продуктов. Если у вас ограничены ресурсы сети или машин (например, надо управлять слабыми машинами-датчиками установленными вне датацентров) то выбирайте CFEngine 3. Если у вас большая инфраструктура и вы сами наполовину разработчик, знающий Ruby, берите Chef. Если вы знаете толк в декларативном подходе и описании состояний – Puppet и Salt – для вас. Кроме того, Salt подойдет тем, кому важна скорость обмена сообщениями и внедрения изменений между узлами инфраструктуры. Ну а ежедневным инструментом для повседневных задач вполне может быть Ansible.

Все инструменты пишутся талантливыми командами и все могут служить верой и правдой вашей инфраструктуре. **Box**

1. Википедия – сравнение CM-систем – http://en.wikipedia.org/wiki/Comparison_of_open_source_configuration_management_software.
2. Методология сравнения – https://www.usenix.org/legacy/event/lisa10/tech/full_papers/Delaet.pdf.

Ключевые слова: управление конфигурацией, CM, Puppet, Chef, SaltStack, Ansible.

Таблица 7. Степень зрелости продукта

CM-инструмент	Освоение	Внедрение	Документация	Коммерческая поддержка	Сообщество	Зрелость
CFEngine 3	2	3	Полная документация, tutorial, справка	Есть	3	5
Puppet	3	4	Полная документация, tutorial, справка	Есть	5	4
Chef	2	3	Полная документация, tutorial, справка	Есть	5	5
SaltStack	4	3	Полная документация, справка	Есть	3	3
Ansible	5	5	Полная документация, tutorial, справка	Есть	3	3



Визитка

АЛЕКСАНДР РУДЕНКО, ЗАО «Молдавская ГРЭС», администратор информационной безопасности, a.rudikk@gmail.com

Облако от VMware

Часть 1. Разворачиваем vCloud Director

Рассмотрим набор продуктов vCloud Suite, его базовые блоки и компоненты, а также установку vCloud Director с описанием всех шагов

Это первая публикация в цикле статей, посвященном vCloud Suite (vCS). vCS – это набор продуктов, включающий все необходимое для построения полноценного, сервисориентированного, адаптивного к любым задачам облака, как частного, так и публичного или гибридного. Основой для этого служит виртуальная инфраструктура VMware vSphere.

Сама по себе тема довольно обширная и сложная с технической точки зрения. В этой работе я лишь немного коснусь теоретических основ и более детально осветю процесс развертывания главного компонента под названием vCloud Director (vCD). Последующие статьи будут посвящены конфигурированию и техническим тонкостям vCD и vCS, а также самим концепциям облачной виртуальной инфраструктуры.

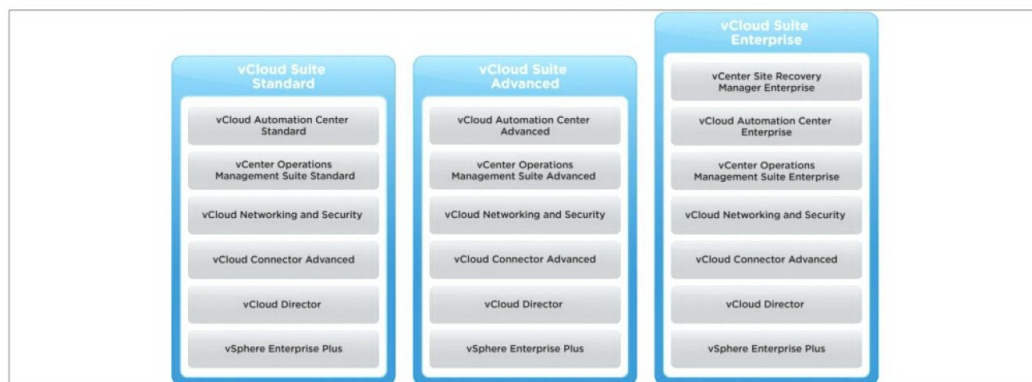
Примечание. В состав vCloud Suite входит несколько отдельных продуктов. Среди них – vCloud Director – главный компонент, являющийся сервером управления облаком. По этой причине я в основном буду говорить именно о vCloud Director и лишь в общих случаях упоминать весь на-

бор продуктов vCloud Suite или его конкретные компоненты.

Пару слов об облаках

К сожалению, все еще приходится начинать с этого вопроса, так как многие считают, что облако – это обычный виртуальный хостинг на OpenVZ, Xen или аналогичных платформах. И это неудивительно, так как часть хостинг-провайдеров, не меняя свою многолетнюю инфраструктуру, попросту добавили к названию «облачный сервис» – дабы быть в тренде. И так, согласно определению [1] NIST (Национальный институт стандартов и технологий) облаком является сервисная модель, которая позволяет, используя удобный сетевой доступ, получить доступ по требованию к общему пулу настраиваемых вычислительных ресурсов, например, сети, серверы, хранилища, приложения или услуги. Получаемые ресурсы могут быть быстро настроены и распределены по своему усмотрению с минимальными усилиями и без необходимости взаимодействовать с поставщиком услуг. Также указаны пять основных критериев облака (са-

Рисунок 1. Сравнение редакций vCloud Suite



мообслуживание, сетевой доступ по требованию, объединение ресурсов, эластичность ресурсов, подсчет только используемых ресурсов).

Исходя из этого классические платформы виртуализации, на базе которых часто предоставляют услуги хостинга, например, VMware vSphere, XenServer, KVM, а также OpenVZ, не являются облачными платформами. Их средства управления требуют серьезной технической подготовки и предназначены для инженеров и специалистов. Также они едва ли соответствуют вышеперечисленным критериям облачной платформы. Именно поэтому существуют такие высокоуровневые платформы, как VMware vCloud Director, CloudStack, OpenStack, Xen Cloud Platform, OpenNebula и прочие, реализующие модель IaaS (Infrastructure as a Service). С помощью vCD можно легко создавать виртуальные дата-центры (vDC) – некий набор вычислительных ресурсов – и предоставлять к ним удобный, интуитивно понятный доступ обычным пользователям-клиентам. И все это с динамическим подсчетом ресурсов, разграничением прав с наращиванием тех или иных мощностей по требованию, а также с автоматизацией большинства типичных задач.

Редакции и лицензирование

Так как vCD – это лишь сервер управления и сам по себе ничего не может, отдельно он не продается и входит в набор vCloud Suite, состав которого выглядит следующим образом:

- vSphere Enterprise Plus** – собственно, основа основ;
- vCloud Director** – «гвоздь сегодняшней программы»;
- vCloud Connector** – механизм, позволяющий соединять облачные инфраструктуры на базе VMware vSphere, как частные, так и публичные;
- vCloud Networking and Security** – новое название vShield Manager – виртуального сетевого экрана для платформы vSphere;
- vCenter Site Recovery Manager** – решение, выполняющее автоматическое аварийное восстановление и тестирование критических ВМ и приложений на резервной площадке;
- vCenter Operations Management** – средство мониторинга и диагностики;
- vFabric Application Director** – средство автоматической настройки и развертывания многоуровневых облачных приложений;

vCloud Automation Center – средство автоматизации различных задач для крупных инфраструктур.

Не вдаваясь в подробности лицензирования, скажу лишь, что получить vCD можно только в составе одной из редакций vCloud Suite. Всего их три: Standart, Advanced и Enterprise. Основные отличия можно увидеть на рис. 1.

Наиболее простым и понятным будет случай, если у вас еще ничего нет. То есть о виртуализации и об облаках вы только задумались. Если так, то покупаете подходящий по функционалу и стоимости пакет vCloud Suite и разворачиваете все с нуля. Если же инфраструктура vSphere уже существует, то придется выполнять обновление лицензии до vCloud Suite по специальной программе. При этом вносить какие-либо изменения в саму инфраструктуру не обязательно. Эта тема выходит за рамки статьи, и ее лучше обсудить со своим ближайшим поставщиком.

Базовые блоки и системные требования

В этом разделе мы более подробно поговорим о необходимых для облачной инфраструктуры базовых блоках. Напомним, что текущая актуальная версия vCD – 5.5.1. Первым делом отмечу, что vCD в отличие от открытых облачных платформ типа CloudStack поддерживает только собственные гипервизоры – ESXi начиная с версии 5.0.

Так как vCD управляет множеством ESXi-хостов по-прежнему с помощью сервера vCenter, то он также необходим. Требуемая версия – не ниже 5.1.

Довольно сложно получить все преимущества от виртуализации, не используя общее сетевое хранилище, поэтому его наличие крайне желательно.

Для обеспечения безопасности и изоляции одних ВМ от других, а также для разделения их сетевого трафика необходим продукт под названием VMware vShield Manage, или vCloud Networking and Security, если говорить по-новому. Если резюмировать вышесказанное, уже вырисовывается базовая инфраструктура VMware vSphere. Причем подойдет не каждая редакция.

Нужна как минимум Enterprise, а если необходимы изолированные сети, то придется обзавестись лицензией на Distributed vSwitch, который доступен только в редакции Enterprise Plus. Собственно, в базовую редакцию vCS входит именно она.

Также для работы vCD необходима база данных (БД). Поддерживаются БД от Microsoft начиная от SQL

Рисунок 2. Настройка vCD. Выбор IP-адреса



Server 2008 Enterprise Edition (желательно с последним пакетом обновлений), а также Oracle 11g. Точные версии можно посмотреть на специальной странице, посвященной совместимости продуктов [2].

Интересно, что БД в инфраструктуре vCD является самым нагруженным узлом. Сервер, исполняющий ее роль, рекомендуется оснастить 16 Гб памяти, четырьмя процессорами и иметь 100 Гб дискового пространства.

Что касается системных требований самого vCD, то они достаточно скромные и сравнимы с требованиями к ресурсам сервера vCenter.

Развертывание

Устанавливается vCD исключительно на следующие дистрибутивы:

- > Red Hat Enterprise Linux 5 (64-bit), Update 4,5,6,8
- > Red Hat Enterprise Linux 6 (64-bit), Update 1,2,3

а также на соответствующие версии CentOS.

Для тех, кому не хочется возиться в консоли Linux, но хочется взглянуть на функционал vCD, есть подготовленная виртуальная машина (Virtual Appliance) от VMware [3]. Развернув его на одном из ESXi-хостов, останется только подключить vCenter и vShield Manager. Стоит отметить, что такая конфигурация не подходит для производственной среды и не поддерживается VMware.

Я же предпочитаю пройти все стадии самостоятельно и лучше разобраться с внутренним устройством продукта. Поэтому будем устанавливать все с нуля. В моем случае использовался дистрибутив CentOS 6.3.

Итак, процесс установки будет состоять из нескольких шагов:

- > подготовка БД;
- > установка базовой ОС;
- > настройка сети базовой ОС;
- > установить недостающие пакеты;
- > установить пакет vCD;
- > создание SSL-сертификатов;

- > запуск постинсталляционного скрипта;
 - > подключение сервера vCenter и vShield.
- Теперь подробнее по каждому из пунктов.

Подготовка БД

В моем случае использовался MS SQL Server 2012 Standard (64-bit), поэтому будет продемонстрирован пример настройки именно его. Требования к БД, а точнее, к ее настройке, довольно специфические. Кроме указанных выше системных требований, необходимо использовать смешанный режим аутентификации, так как vCD не поддерживает Windows-аутентификацию. Также vCD активно использует tmpdb, которая в результате различных операций может значительно вырасти в размере. Рекомендуется разместить файлы этой БД на более быстром массиве с достаточным объемом свободного места.

Создание самой БД вполне стандартно и не будет описываться в этом разделе. Единственное – рекомендуется установить параметры сортировки в Latin1GeneralCS_AS.

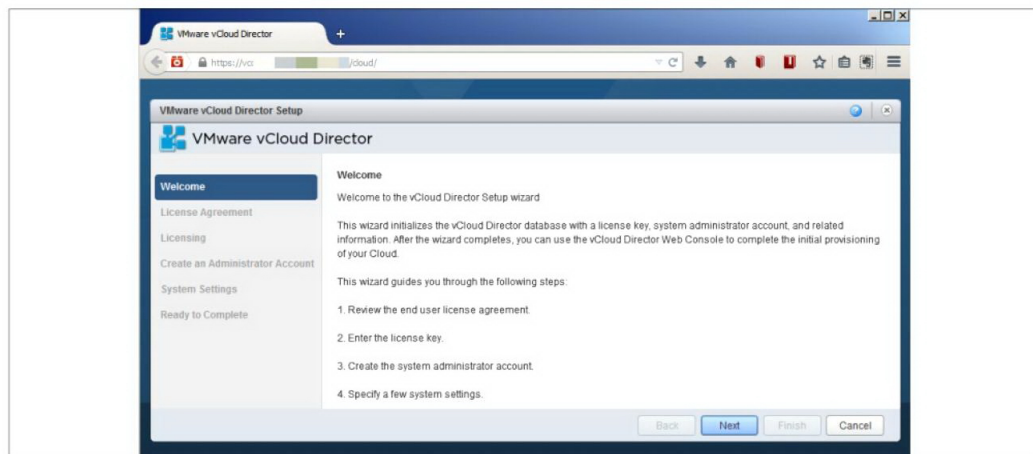
Далее необходимо установить уровень изоляции транзакций для данной БД. Самый быстрый и точный способ это сделать – выполнить следующий запрос:

```
USE [ИМЯ_БД]
GO
ALTER DATABASE [ИМЯ_БД] SET SINGLE_USER
WITH ROLLBACK IMMEDIATE;
ALTER DATABASE [ИМЯ_БД] SET ALLOW_SNAPSHOT_ISOLATION ON;
ALTER DATABASE [ИМЯ_БД] SET READ_COMMITTED_SNAPSHOT ON
WITH NO WAIT;
ALTER DATABASE [ИМЯ_БД] SET MULTI_USER;
GO
```

Теперь необходимо создать пользователя. Можно вручную или с помощью простого запроса:

```
USE [ИМЯ_БД]
GO
CREATE LOGIN [ИМЯ_ПОЛЬЗОВАТЕЛЯ] WITH PASSWORD = 'ПАРОЛЬ',
DEFAULT_DATABASE = [ИМЯ_БД],
```

Рисунок 3. Мастер первичной настройки vCD



```
DEFAULT_LANGUAGE =[us_english], CHECK_POLICY=OFF
GO
CREATE USER [ИМЯ_ПОЛЬЗОВАТЕЛЯ] for LOGIN [ИМЯ_ПОЛЬЗОВАТЕЛЯ]
GO
```

И, наконец, делаем пользователя владельцем созданной выше базы:

```
USE [ИМЯ_БД]
GO
sp_addrolemember [db_owner], [ИМЯ_ПОЛЬЗОВАТЕЛЯ]
GO
```

Приведенные выше запросы помогают выполнить все настройки в считанные секунды, точно и без лишних поисков необходимых параметров.

Установка базовой ОС

Тут нет ничего специфического. Я использовал CentOS 6.3 в минимальной сборке с настройками по умолчанию. Это самый свежий из поддерживаемых vCD дистрибутивов (6.4, 6.5 официально на момент написания статьи не поддерживались). Стоит отметить, что vCD может работать как на физическом сервере, так и на виртуальном. Второй вариант более распространен и является рекомендуемым, поэтому установка CentOS была выполнена на виртуальную машину.

Настройка сети базовой системы

Для работы vCD необходимо иметь как минимум два IP-адреса. На одном будет работать веб-сервер, обслуживающий веб-интерфейс управления vCD. Второй, служебный, используется как своего рода прокси-сервер для перенаправления локальных консолей VM в интерфейс vCD (см. примечание). Конфигурацию с двумя IP можно получить, используя два отдельных сетевых интерфейса или же создав

дополнительный виртуальный интерфейс (Alias) на единственном сетевом адаптере. В случае если базовая ОС установлена на виртуальную машину, предпочтительнее использовать отдельные виртуальные сетевые адаптеры. Если же сервер физический и в наличии лишь один сетевой интерфейс, то решается это следующим образом.

Например, в системе настроен интерфейс eth0. Для того чтобы «повесить» на него еще один IP-адрес, необходимо создать дополнительный конфигурационный файл:

```
# vi /etc/sysconfig/network-scripts/ifcfg-eth0:0
```

со следующим содержанием:

```
DEVICE=eth0:0
ONBOOT=yes
BOOTPROTO=static
IPADDR= <ip-адрес>
NETMASK= <маска-подсети>
```

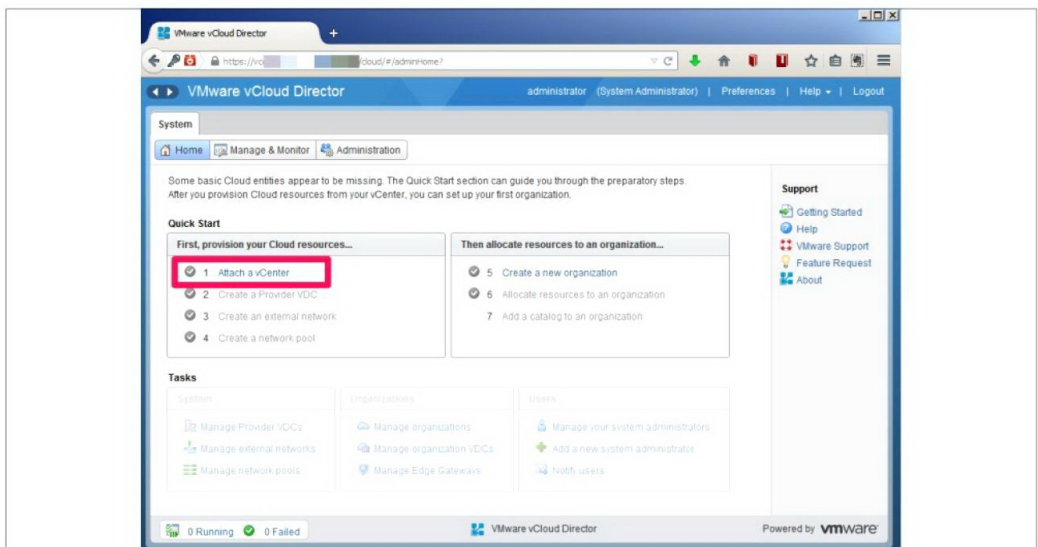
После этого перезагружаем сетевую подсистему.

Примечание. С помощью VMware vSphere Client или ее современной веб-версии можно получить прямой доступ к экрану любой из запущенных VM так, как будто к ней напрямую подключен монитор.

Работая с vCD, пользователь-клиент не имеет доступа к средствам управления vSphere. Все, что у него есть, – это консоль vCD. При этом доступ к своим VM ему также нужен – как минимум для первичной настройки сети и разрешения удаленного доступа. Для этого и используется консоль-прокси на vCD с отдельным IP. Через него пользователю доставляется консоль VM прямо в интерфейсе vCD.

```
# service network restart
```

Рисунок 4. Мастер подключения к vCenter и vShield Manager



и видим в системе еще один – псевдоинтерфейс (eth0:0). Проверить его работоспособность можно с помощью пинга с другого сервера.

Кроме этого, необходимо создать DNS-записи для каждого из этих IP-адресов. Например, vcd01.company.com для консоли управления и vcd01-rc.company.com для проксирования консолей ВМ. Перед переходом к следующему шагу необходимо убедиться, что DNS возвращает правильные адреса и имена.

Установка недостающих пакетов

Если запустить установку vCD в неподготовленной системе, то установщик быстро завершит работу, указав список недостающих пакетов. Их все можно установить одной командой из стандартных репозиторий:

```
# yum -y install alsa-lib libICE libSM libX11 libXau \
libXdmcp libXext libXi libXt libXtst \
pciutils redhat-lsb
```

Устанавливаем пакеты vCD

Теперь необходимо загрузить установочный пакет самого vCD [4]. Для этого потребуются зарегистрироваться или войти, если учетная запись VMware уже существует. Когда загрузка будет завершена и установочный пакет будет перемещен на наш сервер, можно приступить к установке:

```
# bash <путь_к_пакету>/ \
vmware-vcloud-director-5.5.1-1654672.bin
```

Хотя CentOS 6.3 является официально поддерживаемым дистрибутивом, инсталлятор выдаст запрос на подтверждение установки, заявив обратное. Видимо, ему хочется именно RHEL. В остальном установка пройдет быстро и без вопросов.

Создание самоподписанных SSL-сертификатов

Сертификаты обеспечивают максимальный уровень надежности сетевых взаимодействий с vCD. Каждый сервер должен иметь как минимум два сертификата: по одному для каждого его IP-адреса.

Создание сертификата для управляющего интерфейса:

```
# /opt/vmware/cloud-director/jre/bin ./keytool -keystore \
/etc/certificates.ks -storetype JCEKS -storepass \
<ПАРОЛЬ> -genkey -keyalg RSA -alias http
```

Далее будет необходимо ответить на ряд стандартных для этой процедуры вопросов, таких как имя организации, город и так далее.

И аналогично для адреса прокси:

```
./keytool -keystore /etc/certificates.ks \
-storetype JCEKS -s torepass \
<ПАРОЛЬ> -genkey -keyalg RSA -alias consoleproxy
```

Здесь будут те же вопросы, ответы на которые можно давать аналогичные.

Запуск конфигурационного скрипта

Теперь все готово, и можно приступить к конфигурированию vCD:

```
# bash /opt/vmware/vcloud-director/bin/configure
```

Настройка vCD производится в формате вопрос-ответ. Всего несколько вопросов.

На первом шаге необходимо указать IP-адрес из списка доступных, на котором будет работать веб-сервер (см. рис. 2).

Далее из оставшегося списка необходимо выбрать IP-адрес для прокси. На третьем шаге нужно указать путь к хранилищу сертификатов, который мы задали на предыдущем шаге при их создании (в моем случае /etc/certificates.ks). Если хотим отправлять логи на сервер SYSLOG, то указываем его параметры.

И, наконец, выбираем из предложенных вариантов, какой тип БД будем использовать.

Далее вводим параметры подключения БД, адрес сервера, имя базы, имя пользователя и пароль. Все должно пройти гладко.

В итоге будет предложено запустить службы vCD и войти браузером по указанному адресу.

Подключение сервера vCenter и vShield

При первом входе в веб-консоль vCD вас встретит мастер первичной настройки (см. рис. 3). Здесь необходимо принять условия лицензии, ввести ключ продукта и задать пароль к учетной записи администратора.

Далее придется выполнить вход под учетной записью Administrator, используя указанный выше пароль, и запустить мастер подключения к vCenter и vShield Manager (см. рис. 4). Более детально все эти процессы будут рассмотрены в следующей статье.

Как уже было сказано в первых разделах статьи, vCD – это лишь средство управления высокого уровня. Его применение возможно только при наличии базовой (как минимум) инфраструктуры vSphere (ESXi, vCenter, vShield, общее хранилище), причем все это уже должно быть настроено и работать.

В этой статье продемонстрирована установка первого сервера vCD в группе. Подключив к нему ресурсы vSphere (vCenter, vShield), мы получим, в терминологии VMware, ячейку (vCD Cell).

Облако может состоять всего из одной такой ячейки или из множества (несколько vCD в группе). Ячейки могут быть разнесены территориально или географически, но подключаются к единой БД и действуют как единое целое. Архитектура, а также техническим особенностям vCD будут посвящены следующие статьи. **EOF**

1. Определение облака – www.nist.gov/it/cloud.
2. Совместимость продуктов VMware – www.partnerweb.vmware.com/comp_guide2/sim/interop_matrix.php.
3. Страница загрузки vCD Virtual Appliance – www.vmware.com/go/try-vcloud-director.
4. Страница загрузки vCD – www.vmware.com/go/download/vcloud-director.

Ключевые слова: виртуализация, VMware vCloud Director, vSphere, Cloud.

**IX Всероссийский форум для intranet-специалистов
и руководителей отделов внутренних коммуникаций**

**УЧАСТВУЙТЕ
И ОЦЕНИВАЙТЕ!**
конкурс intranet-порталов
BEST INTRANET RUSSIA

Best Intranet 9th RUSSIA-2014

**Как сделать intranet-портал эффективным инструментом
для решения бизнес-задач компании?**

18-20 июня 2014
Holiday Inn Moscow Sushevsky

СРЕДИ ДОКЛАДЧИКОВ ФОРУМА:



Александр
ЩЕГЛОВ,
СЕВЕРСТАЛЬ



Елена
МУРАТШИНА,
РОССЕЛХОЗ-
БАНК



Анастасия
КОЖНИКОВА,
FORD SOLLERS



Анна
НЕСМЕВА,
Независимый
консультант



Ирина
ЛЕВИТИНА,
NORDEA
БАНК



Юлия
ЭЛОТНИК,
МАКДОНАЛДС



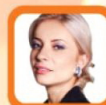
Андрей
КОЧЕРОВ,
МЕЧЕЛ



Алина
МОРОЗОВА,
ЛЕТО БАНК



Екатерина
ЛЕЖНЕВА,
МТС



Светлана
ДАНИЛОВА,
МДМ БАНК

**СПЕЦИАЛЬНЫЙ ДЕНЬ
(18 ИЮНЯ 2014 Г.)**
Аудит intranet-портала:
первый шаг к оптимиза-
ции и раскрутке

КОМУ СТОИТ ПОСЕТИТЬ ФОРУМ?

- Директору по внутренним коммуникациям;
- Специалисту по внутренним коммуникациям;
- Менеджеру по внутренним коммуникациям;
- Редактору Intranet-портала;
- Директору по персоналу;
- Заместителю директора по персоналу;
- PR-директору;
- HR-директору;
- Директору по организационному развитию

КЛЮЧЕВЫЕ ВОПРОСЫ ФОРУМА:

- Как эффективно использовать intranet-портал для повышения уровня вовлеченности сотрудников во внутренние бизнес-процессы компании?
- Как управлять информационным содержанием для его популяризации в компании?
- В чем преимущества социальной сети перед внутренним корпоративным порталом в компании?
- Как HR-служба может использовать intranet-портал в качестве инструмента решения своих задач?
- Какие технологические аспекты поддержки и модернизации intranet-портала существуют в современной компании?

прими компетентное решение®
Quorum

www.globalforumfactory.ru/INT14



Визитка

ПАВЕЛ ЗАКЛЯКОВ, IT-специалист, amd7@mail.ru

USB_ModeSwitch, или «Кот в мешке»

Подключили USB-сотовый модем к ОС Linux, а он не заработал? Мы расскажем, почему это случилось и что делать?

Многим нравится подход: «подключил и работай», именно так позиционируются популярные на рынке решения в виде USB-сотовых модемов. Они используются не только с ноутбуками как основной канал выхода в Интернет (там, где нет Wi-Fi или проводного доступа), но и на серверном оборудовании в качестве резервного канала управления, шлюза для IP-телефонии [1, 2] или для SMS-оповещения [3]. Администраторы ОС Linux могут столкнуться с тем, что не все подобные устройства определяются ядром ОС, даже если их чипсет в списке поддерживаемых.

Первопричина проблемы

Поскольку классификация и стандартизация аппаратных устройств (в том числе и их функций) нередко отстаёт от жизни, что-то «новое» может не поддерживаться той или иной ОС. С этой целью производители ОС документируют API драйверов и предлагают разработчикам самостоятельно реализовывать недостающую функциональность.

Рисунок 1. USB-модем сотовой связи (Модель ZTE MF192)



Проблема носителя

Программные доработки обычно записываются на внешний носитель и поставляются вместе с устройством. Когда-то это были дискеты с драйверами, позже их вытеснили оптические диски.

Поскольку ёмкость носителей чаще избыточна, то на них можно встретить различное вспомогательное программное обеспечение, документацию. От них можно было бы и отказаться, выложив всё необходимое в сети Интернет, но вряд ли вам понравится ситуация, когда будет нужна программа от модема, с помощью которого планируете подсоединяться к этой самой сети, или фирма перестала поддерживать оборудование или закрылась.

Если следовать прогрессу, то драйвер, его исходные коды, документацию следовало бы записать на флешку и вручить её покупателю вместе с основной покупкой.

Проблема очередности

Во избежание лишних вопросов от пользователей к службе поддержки во многих коммерческих ОС требуется соблюдать очередность, а именно, чтобы сначала было установлено необходимое ПО, а потом уже подключалось новое оборудование.

Напрашивается мысль, что разработчики Windows специально придумали себе эту проблему, чтобы с гордостью её решить позднее. Поскольку пользователи указанной ОС занимают достаточный процент рынка, для них в корне файловой системы накопителя может быть записан текстовый файл autorun.inf, указывающий, какие программы ОС следует запустить.

Решение проблем: 2-в-1

Один из вариантов решения проблем, описанных выше, состоит в том, чтобы объединить функциональность носителя и устройства в одном корпусе.

Поскольку шина USB широко распространена, под носителем будем понимать широкий класс различных устройств, объединённых спецификацией USB mass storage device class, или коротко Mass Storage.

Использование устройств данного класса позволяет обойтись штатными драйверами (возможностями) ОС, без уточнения, что есть носитель – флеш-память, оптический привод или что-то ещё. Такой подход совместно с дешёвой производством позволил сделать данные устройства стандартом по обмену информацией.

Для того чтобы носитель с драйвером оказался подключённым первым, в устройство можно добавить программно-управляемый переключатель. Собственно, такое решение и используется в устройстве, приведённом на фотографии ниже (см. рис. 1), за тем лишь исключением, что флеш-накопитель эмулирует работу внешнего привода CD-ROM.

Сотовые модемы, подобные приведённому выше, сегодня стоят около 500 руб. и устроены, как показано на рис. 2.

При первом подключении устройства к компьютеру, внутренний блок управления (см. рис. 2) подключает виртуальный CD/DVD-накопитель с диском. Включается режим CD/DVD-привода (на рисунке цифра 1).

Далее, по задумке производителей, ОС обнаруживает подключённый CD/DVD-привод с диском. ОС не подозревает, что её «обманывают», и честно производит просмотр файлов на носителе. Она находит файл автозапуска `autorun.inf` в корне виртуального носителя, считывает имя файла-установщика и запускает его. Таким образом, управление передаётся программе установщика драйвера, документации и другого необходимого ПО-устройства для данной ОС.

По завершении процесса установленное ПО запускается и подаёт специальную команду устройству (обычно на виртуальный CD/DVD об извлечении диска), что воспринимается блоком управления устройства как команда отключить виртуальный CD/DVD-привод и подключить сотовый модем. Включается режим модема (на рисунке цифра 2). Поскольку к этому моменту ОС уже имеет представление о том, какой модем будет подключен и как его настраивать, то она это и делает, предоставляя в последующем пользователю всю необходимую функциональность.

При повторном подключении устройства оно также понимается ОС как виртуальный CD/DVD-привод, но, так как необходимое ПО модема уже установлено в системе, оно успевает отследить ситуацию быстрее, чем это делает пользователь, и подать команду на включение режима модема. Таким образом, пользователь ничего не замечает и работает с устройством, как он этого и ожидает: «подключил и работает».

Всё бы хорошо, но разные ОС имеют разный формат исполняемых файлов (и API-драйверов), а производители оборудования не стремятся отслеживать все нюансы. Таким образом, сделав добро пользователям одной ОС, они ставят пользователей других ОС в неудобное положение. А именно: купил устройство, подключил, а оно не работает. Точнее, работает, но не так, как надо.

В системе отображается внешний USB-CD/DVD-привод с диском (где лежат программы для другой, не совместимой с вашей ОС), а модема нет. Это обидно, особенно если ядром ОС чипсет модема поддерживается. Хорошо, если вы догадаетесь извлечь виртуальный носитель из виртуального привода средствами пользовательского интерфейса, и это поможет (то есть произойдет переключение режима работы устройства). А если нет?

Полагаю, что несколько лет назад (предположительно, в сентябре 2009 года, так как в это время сделана первая запись в файле `ChangeLog`) именно в такой ситуации и оказался пользователь ОС Linux Джошуа Дитц (Josua Dietze), после чего он и придумал решение – написать небольшую программу для ОС Linux, которая могла бы переключать его USB-модем из одного режима в другой. Имя этой программы сегодня `USB_ModeSwitch` [4].

Позднее для удобства программу разбили на два пакета (две логические части):

usb_modeswitch – содержит необходимые программы;

usb_modeswitch-data – содержит информацию о поддерживаемых устройствах.

Если они не входят в базовый репозиторий вашего дистрибутива Linux, то их можно найти в альтернативных. Например, для Red Hat Linux (и совместимых) это может быть `RepoForge` [5]. Также возможна установка из исходных кодов, доступных на сайте проекта.

С момента зарождения версии 0.1 по сегодняшний день программа развивалась, неоднократно улучшалась, заинтересовала собой не одного программиста, благодаря чему сегодня она поддерживает более 300 идентификаторов USB-устройств. И это число продолжает расти благодаря письмам пользователей. Может, и вы, читатель, сможете помочь проекту.

Отдельное спасибо Алексею Дмитриеву, осуществившему перевод страницы проекта на русский язык [6].

ZeroCD(TM) резюме проблемы

Таким образом, всё больше USB-устройств (особенно из линейки высокоскоростных WAN-устройств, созданных на основе сотовых чипсетов) имеют встроенные драйверы для ОС Windows; будучи подключёнными к компьютеру в первый раз, они ведут себя как USB-флеш-накопители и начинают установку своего программного обеспечения. По окончании установки (и при всех последующих подключениях) драйвер переключает режим гаджета, накопитель исчезает, и можно увидеть новое устройство (например, USB-модем).

По мнению общественности [7], такой механизм переключения устройств получил название «ZeroCD(TM)» так как он позволяет обойтись без отдельного носителя с драйверами. Значок (TM) принадлежит фирме-произ-

Рисунок 2. Внутреннее функциональное устройство USB-модема сотовой связи



Замечание 1

Для перехвата команд и сообщений, проходящих по USB-шине компьютера в ОС Windows можно использовать программы USBPcap и Wireshark [12], увидеть результат можно, например, запустив из консоли следующие команды:

```
cd C:\Program Files\USBPcap
USBPcapCMD.exe -d \\.\USBPcap2 -o - | .\
"C:\Program Files\Wireshark\Wireshark.exe" -k -i -
```

водителю модемов Option, придумавшей данный термин и начавшей использовать данную функцию в своих устройствах.

Поддержка устройств в ОС Linux

Как вы понимаете, говорить ни о какой документации о поддержке со стороны ОС Linux пока что не приходилось. С другой стороны, большинство USB-сотовых модемов могли нормально работать в обоих режимах (при поддержке со стороны ядра ОС Linux).

Для поддержки первого режима (накопитель) достаточно собрать ядро, включив опцию CONFIG_USB_STORAGE. За второй режим отвечают CONFIG_USB_SERIAL_GENERIC, CONFIG_USB_SERIAL_PL2303, CONFIG_USB_SERIAL_PL2303, CONFIG_USB_SERIAL_OPTION (улучшенный драйвер последовательного порта, стандарт Linux для скоростных 3G/4G модемов) и др.

Таким образом, для работы «специфичных» USB-устройств, представляющих собой «кота в мешке», при первом (и последующих, без программной поддержки со стороны ОС) подключении необходимо было лишь решить проблему переключения их режима работы из режима накопителя в режим модема. То есть необходимо было научить ОС Linux переключить устройства так же, как это делает программа (драйвер модема) в ОС Windows. К счастью, даже при отсутствии исходных кодов для проприетарного драйвера обмен данными между ПО на стороне ОС и аппаратным устройством по шине USB идёт в открытом виде.

Используя библиотеку языка Си для стандартных функций USB-устройств libusb [8] и программы мониторинга USB-шины [9-12], возможно записать в текстовый файл все прошедшие «высокоуровневые сообщения», посылаемые устройству, а после визуально сопоставить момент переключения устройства с перехваченными данными. У большинства программистов, даже поверхностно знакомых с USB-протоколом, задача выделить команду или действие, переключающее режим, не составит труда.

Последующее воспроизведение подобного сценария переключения по правилам ОС Linux (или ОС UNIX) не проблема.

По опыту Дейза (Josua Dietze) [13] переключение чаще всего происходит командой eject для SCSI-устройств, что то же самое есть последовательность байтов «55 53 42 43 12 34 56 78 00 00 00 00 00 06 1b 00 00 00 02 00 00 00 00 00 00 00 00 00 00», хотя могут быть и другие последовательности.

Каждый раз «переключать» устройство вручную даже самым упорным пользователям надоеет, поэтому рассмотрим, как можно автоматизировать данный процесс с помощью USB_ModeSwitch.

Установка usb_modeswitch

Программа постоянно обновляется (актуальная 1.1.0), так что ее автор настоятельно рекомендует убедиться в отсутствии старой версии, которую следует удалить перед установкой новой. Для установки в Red Hat Linux (и совместимых) скачайте пакеты RepoForge и установите их командами:

```
# rpm -ihv usb_modeswitch-1.2.3-1.el6.rf.x86_64.rpm
# rpm -ihv usb_modeswitch-data-20120120-1.el6.rf.noarch.rpm
```

Для установки из исходных текстов потребуются наличие скриптового языка tcl.

Далее:

```
$ wget http://www.draisberghof.de/usb_modeswitch/ .\
usb_modeswitch-2.1.1.tar.bz2
$ tar -xvf usb_modeswitch-2.1.1.tar.bz2
$ cd usb_modeswitch-2.1.1
# make install
```

Последняя команда установит: shell-скрипт оболочки, небольшой tclsh-скрипт для udev (для него и нужен tcl либо jimsh), конфигурационный файл, man-страницу документации и скомпилированный исполняемый файл.

Замечание. Есть и другие способы установки, не требующие языка tcl (подробнее см. файл README внутри архива). Аналогичным образом ставится пакет usb_modeswitch-data с данными о различных USB-устройствах.

В процессе установки конфигурационные файлы копируются в /usr/share/usb_modeswitch, а файл правил udev – в /lib/udev/rules.d. Каталог /etc/usb_modeswitch.d, использовавшийся в более ранних версиях, сейчас зарезервирован для пользовательских конфигурационных файлов.

После установки программы и её базы устройств всё готово для работы; если ваш USB-модем есть в базе данных, то сможете использовать его сразу после подключения. Если же устройство не работает, как следует, то есть «подключил и работает» не получилось, не расстраивайтесь, ниже приводится несколько советов по выявлению причин возникновения проблемы и советы по её устранению.

Если «подключил и не работает»

Для работы в ручном режиме потребуется запуск usb_modeswitch. Для получения положительного результата можно двигаться по двум направлениям: вводить команды непосредственно из командной строки либо редактировать конфигурационный файл. В качестве отправной точки для создания собственной конфигурации можете взять примеры из файла device_reference. Он содержит много полезных комментариев и подсказок.

Пользовательский конфигурационный файл может иметь любое допустимое ОС имя и располагаться в любом каталоге, который доступен для чтения. Ручной режим предназначен для тестирования и исследований. Подробнее об этом написано ниже.

Внутренние процессы

Если USB-модем поддерживается, но не работает, первым делом включите журналирование, как это описано в разделе «Диагностика (эксперименты)», далее по тексту.

Для тестирования, диагностики и «приручения» нового «гаджета» более чем достаточно существующей функциональности программы USB_ModeSwitch при её запуске в ручном режиме. Будут ли параметры запуска задаваться конфигурационным файлом или непосредственно из командной строки, неважно.

Чтобы получить список возможных параметров команды, запустите:

```
usb-modeswitch -h
```

Конфигурационный файл задаётся параметром `-c`. Следует иметь в виду, что отсутствие или наличие ряда других опций может влиять на действие ключа `-c` (подробнее о параметрах запуска см. [6] или документацию к программе).

Конфигурационный файл проще создавать не с нуля, а взять любой готовый из `/usr/share/usb_modeswitch` и отредактировать. В файле `device_reference.txt` имеются подсказки по различным семействам модемов с объяснением ряда их параметров.

Замечание. Команду `usb_modeswitch`, как и все программы, использующие `libusb`, следует запускать в командной строке от имени суперпользователя (или через `sudo`).

При работе в автоматическом режиме действуют следующие компоненты, перечисленные в порядке вызова:

`/lib/udev/rules.d/40-usb_modeswitch.rules` – `udev`-правила, запускающие оболочку при подключении устройства с указанным ID (`vendor/product`). Если устройство после переключения режима создаёт последовательные порты, второе правило снова запускает оболочку и создаёт символическую ссылку на правильный порт (см. ниже).

`/lib/udev/usb_modeswitch` – `shell`-скрипт, запускаемый программой. Начиная с версии 1.1.6 он полностью совместим с `dash`, применяемым в `Ubuntu`, так же как и с более старыми версиями `bash`.

`/usr/sbin/usb_modeswitch_dispatcher` – проводит дополнительную проверку устройства и непосредственно переключает режим, руководствуясь конфигурационным файлом для конкретных `vendorID` и `productID` USB-устройства. Если после переключения режима не нашлось модуля (кода в ядре) для обеспечения работы устройства, то диспетчер попытается использовать код ядра, включённый при конфигурировании опций `CONFIG_USB_SERIAL_OPTION`.

`/etc/usb_modeswitch.conf` – общий конфигурационный файл для включения функции журналирования (при диагностике) или для запрета переключения режимов (обычно для того, чтобы получить доступ к виртуальному флеш-накопителю).

`/usr/share/usb_modeswitch` – каталог, содержащий индивидуальные файлы для каждого USB-устройства, именванные в соответствии с их ID (или другими идентификаторами в случае совпадения ID). Если в нем имеется файл, совпадающий по имени с идентификаторами вашего устройства (которые можете определить по журналу `/var/log/messages` либо утилитой `lsusb`), все шансы за то, что оно поддерживается, даже если его модель (указанная в комментариях файла) не совпадает.

`/usr/sbin/usb_modeswitch` – собственно двоичный исполняемый файл, производящий переключение режимов.

Замечание 2

Если к компьютеру с ОС `Linux` подключить USB-модем, изображённый на рис. 1, то после подключения в файле `/var/log/messages` можно увидеть:

```
May 19 21:34:42 kernel: usb 2-1.2: new high speed USB device
number 5 using ehci_hcd
May 19 21:34:42 kernel: usb 2-1.2: New USB device found,
idVendor=19d2, idProduct=1514
May 19 21:34:42 kernel: usb 2-1.2: New USB device strings: Mfr=2,
Product=1, SerialNumber=3
May 19 21:34:42 kernel: usb 2-1.2: Product: MF192
May 19 21:34:42 kernel: usb 2-1.2: Manufacturer: ZTE
May 19 21:34:42 kernel: usb 2-1.2: SerialNumber:
*****
May 19 21:34:42 kernel: usb 2-1.2: configuration #1 chosen from 1
choice
May 19 21:34:42 kernel: Initializing USB Mass Storage driver...
May 19 21:34:42 kernel: scsi 5:0:0:0: CD-ROM          ZTE
Storage devices
May 19 21:34:42 kernel: usbcore: registered new interface driver
usb-storage
May 19 21:34:42 kernel: USB Mass Storage support registered.
May 19 21:34:43 kernel: scsi 5:0:0:0: CD-ROM          ZTE
Datacard CD-ROM 0001 PQ: 0 ANSI: 0
May 19 21:34:43 kernel: scsi 5:0:0:0: Attached scsi generic sgl
type 5
May 19 21:34:43 kernel: sr0: scsi3-mmc drive: 0x/0x caddy
May 19 21:34:43 kernel: Uniform CD-ROM driver Revision: 3.20
May 19 21:34:43 kernel: sr0: CDROM (ioctl) error, command:
Xpwrite, Read disk info 51 00 00 00 00 00 00 02 00
```

Совет: для просмотра файла журнала «живую» можно открыть ещё одну консоль и в ней набрать команду:

```
tail -f /var/log/messages
```

Ядром ОС было обнаружено новое USB-устройство с идентификаторами производителя `19d2` и моделью устройства `1514`. Далее при установленном, но не работающем пакете `usb_modeswitch` переключение режима работы устройства можно выполнить вручную, для выполнения команды:

```
# usb_modeswitch -W -v 19d2 -p 1514 -M J
"55534243123456780000000000000061b000000020000000000000000000000"
```

Процесс переключения режима работы устройства в журнале системы (`/var/log/messages`) можно обнаружить по появлению устройства с новым USB-идентификатором и подключению ядром псевдофайлов-устройств (`/dev/ttyACM0` и др.) для работы с последовательным портом модема:

```
May 19 21:38:03 kernel: usb 2-1.2: new high speed USB device
number 9 using ehci_hcd
May 19 21:38:03 kernel: usb 2-1.2: New USB device found,
idVendor=19d2, idProduct=1515
May 19 21:38:03 kernel: usb 2-1.2: New USB device strings: Mfr=2,
Product=1, SerialNumber=3
May 19 21:38:03 kernel: usb 2-1.2: Product: MF192
May 19 21:38:03 kernel: usb 2-1.2: Manufacturer: ZTE
May 19 21:38:03 kernel: usb 2-1.2: SerialNumber:
*****
May 19 21:38:03 kernel: usb 2-1.2: configuration #1 chosen from 1
choice
May 19 21:38:03 kernel: cdc_acm 2-1.2.1.0: ttyACM0: USB ACM device
May 19 21:38:03 kernel: cdc_acm 2-1.2.1.2: ttyACM1: USB ACM device
May 19 21:38:03 kernel: scsi9 : SCSI emulation for USB Mass
Storage devices
May 19 21:38:03 kernel: cdc_acm 2-1.2.1.5: ttyACM2: USB ACM device
May 19 21:38:03 localhost modem-manager: (ttyACM0) opening serial
device...
```

После можно использовать стандартное программное обеспечение для коммутируемого (`dial-up`) доступа (например, консольную утилиту `wvdial`), как если бы у вас был привычный `hayes`-совместимый модем, например, на порту `/dev/ttyACM0` (по аналогии с `COM`-портом – `/dev/ttyS0`). (Для разных моделей модемов и версий ядра именование псевдофайла последовательного порта устройства может отличаться.)

Следующий шаг за ОС

После того как режим работы USB-устройства переключен и драйвер последовательного порта запущен, наступает очередь операционной системы обнаружить новое устройство. Собственно, это уже не относится к теме статьи, но скорее всего понадобится читателям, так как по большей части использование USB_ModeSwitch – лишь один из шагов к достижению более высокой цели – подключению к сети сотового оператора.

На следующем шаге возможны проблемы у программы NetworkManager (или его компоненты ModemManager), которая до сих пор часто ошибается при определении правильного порта. В связи с этим можно рекомендовать использовать другие программы, чья настройка проще, например, wvdial, UMTSmon, как и обеспечивающие графический интерфейс для запуска PPP, например, kppp; правда, они требуют некоторых знаний.

Sakis3G

Отдельно хочется отметить хорошую и самодостаточную программу Sakis3G, которая сокращала пользователям утомительный путь настройки различных компонентов для подключения беспроводной связи и содержала в своем составе, среди прочего, последнюю сборку USB_ModeSwitch. К сожалению, на момент написания статьи, сайт поддержки программы уже недоступен, и вряд ли стоит ожидать выхода новых версий.

Ознакомиться как со старыми версиями программы (которые функциональны и по сей день), так и с бывшим сайтом проекта можно в архиве Интернета [14], некоторые компоненты можно найти в [15].

Откуда берутся проблемы?

Главное препятствие полностью автоматического запуска вновь подключенного модема программой NetworkManager и иже с ним – это определение правильного порта. Зачастую, после переключения режима, создается более одного порта.

На врезке выше красным выделены три порта ttyACM0, ttyACM1, ttyACM2, а в ряде устройств их может быть до пяти. К сожалению, не все они являются действительно последовательными портами. Проблема в том, что программа, желающая установить соединение через модем, не может понять, какие из портов есть правильные. Простой перебор портов в надежде соединиться хотя бы через один из них, к сожалению, не даёт 100% верный результат, так как «ложные» порты работают, как настоящие, и даже отвечают на некоторые Hayes-совместимые AT-команды, но при этом не инициируют соединение. Только правильный порт является точкой передачи данных.

Как достоверно узнать, какой из созданных портов есть правильный и зачем производители придумали «ложные», автору неизвестно. (Если вы знаете ответ – добро пожаловать на форум журнала.) Перебрать все порты и найти верный несложно, но, к сожалению, NetworkManager руководствуется своими критериями проверки и поэтому временами ошибается, зато упомянутый выше Sakis3G, находит истинный порт играючи.

Чтобы не разбираться каждый раз, какой порт верный: ttyACM2, ttyACM3 или другой, начиная с версии 1.1.2 usb_

modeswitch добавляет мягкую (символическую) ссылку на первично правильно определённый порт, обеспечивающий передачу данных, если устройство создаёт последовательные порты. Эта ссылка носит имя /dev/gsmmodem (с соответствующим номером, в случае подключения нескольких устройств).

То есть пользовательские программы могут всегда обращаться к необходимому устройству по одной и той же ссылке /dev/gsmmodem, даже если вы вдруг поменяете свой USB-сотовый модем с одного на совсем другой. Вы можете пользоваться ссылкой при настройке программ дозвона типа wvdial и других.

Не дайте наступить на ваши грабли другому

Если ваш USB-модем или иное устройство не поддерживались изначально, а вам удалось его запустить вручную, переключив режим, не поленитесь прописать соответствующее udev-правило, конфигурационный файл и сообщить об этом разработчикам USB_ModeSwitch, чтобы они пополнили базу известных USB-устройств и другие пользователи могли не наступать на ваши грабли повторно.

Поскольку проект usb_modeswitch некоммерческий, то назначить кого-то ответственным за жёсткое тестирование всех присланных правил невозможно. Иными словами, в базе устройств могут содержаться ошибки и неточности. Не удивляйтесь, если ваше устройство с какими-то конкретными vendorID и productID содержится в списке поддерживаемых, но не переключается. Постарайтесь найти ошибку, переключая устройство в ручном режиме. Для наиболее продвинутых пользователей ниже приводится несколько советов.

Хороший пример настройки модема Alltel UM175AL USB EVDO под ОС Linux опубликовал Марк Зиезмер (Mark A. Ziesemer) в [16].

О других способах переключения режима

Кроме рассмотренного ранее способа переключения устройства штатной командой `eject`–«извлечь» (ответственную за данную команду последовательность и пример использования см. выше во врезке), могут быть и другие варианты.

Так, на устройство можно послать одну из «вендор-специфичных» управляющих команд. К сожалению, её надо либо знать заранее, либо искать последовательности на форумах, либо использовать USB-сниффер под ОС Windows, для которой есть драйвер устройства, либо заняться переспективным перебором.

В некоторых устройствах можно насильно отключать модуль поддержки накопителя, что расценивается как команда `eject`, рассмотренная выше.

Наконец, обновите базу поддерживаемых устройств, зайдите на форум [17]. Может, устройство с вашими vendorID и productID уже кто-то протестировал и внёс в список поддерживаемых, пока вы читаете эти строки?

Известные проблемы

Существует ряд известных автору проекта USB_ModeSwitch проблем: с устройствами с ID 05c6:1000 и 19d2:2000, с ядрами до 2.6.26, с некоторыми модемами ZTE и со старыми ОС. Так как они затрагивают лишь узкий круг читателей, позна-

комиться с ними вы можете на сайте проекта или на форуме [17].

В рамках данной статьи мы поверхностно познакомились с внутренним устройством большинства современных сотовых модемов, с проблемой переключения режима работы устройств и с тем, как с ней справляется программа `usb_modeswitch` в ОС Linux.

Материал может быть полезен системным администраторам, так как указанные устройства могут использоваться и на серверном оборудовании для различных SMS-уведомлений, в качестве шлюза для телефонии, а также для обеспечения резервного интернет-канала. **EOF**

1. FAQ for Huawei 3G USB modem and Asterisk – <http://wiki.e1550.mobi>.
2. Лето, отпуск, asterisk, или Сам себе VoIP-оператор – <http://habrahabr.ru/post/185250>.
3. SMS Server Tools 3 – <http://smstools3.kekekasvi.com>.
4. Josua Dietze USB_ModeSwitch: Handling Mode-Switching USB Devices on Linux – http://www.draisberghof.de/usb_modeswitch.
5. RepoForge – альтернативный репозиторий (полезные RPM-пакеты) – <http://repoforge.org>.
6. Управление бистабильными USB-устройствами под Linux./Пер. с англ.[1] Алексея Дмитриева.
7. О режиме ZeroCD (TM) – http://en.wikipedia.org/wiki/Virtual_CD-ROM_switching_utility, <http://www.hub.ru/wiki/ZeroCD>.
8. Библиотека языка Си libusb, дающая приложениям доступ к USB-устройствам в разных ОС – <http://www.libusb.org>.
9. USB-сниффер USBPcap – USB Packet capture for Windows <http://desowin.org/usbpcap>.
10. USB-сниффер usbsnoop – <http://usbsnoop.sourceforge.net>, <http://benoit.papillaud.free.fr/usbsnoop>.
11. Коммерческий USB-сниффер USBlyzer, тестовая версия на 33 дня – <http://www.usblyzer.com/download.htm>.
12. Настройка Wireshark для анализа USB – <http://wiki.wireshark.org/CaptureSetup/USB>.
13. Josua рекомендует использовать команду `eject` – http://www.draisberghof.de/usb_modeswitch/bb/viewtopic.php?f=3&t=1393&p=9353&hilit=eject.
14. Архив сайта Sakis3G – <https://web.archive.org/web/20130511202305/http://www.sakis3g.org>.
15. Компоненты Sakis3G – <https://aur.archlinux.org/packages/sakis3g>.
16. Настройка модема Alltel UM175AL USB EVDO под ОС Linux Ubuntu – <http://blogger.ziesemer.com/2008/10/alltel-um175al-usb-evdo-ubuntu.html>.
17. Форум проекта `usb_modeswitch` – http://www.draisberghof.de/usb_modeswitch/bb.

Ключевые слова: USB, модем, Linux, установка драйвера, `usb_modeswitch`.



Компоненты: 121 карта работы,
79 карт премий, 30 жетонов, 1 токен, правила

Системный
администратор

games.samag.ru



Визитка

РАШИД АЧИЛОВ, главный специалист по защите информации в компании, занимающейся автоматизацией горнодобывающей промышленности, shelton@sheltonsoft.ru

Большой Брат видит тебя

Программы для скрытого наблюдения за пользователем, позволяющие перехватывать его почту и символы, вводимые с клавиатуры, контролирующие использование сменных носителей и делающие скриншоты его экрана, записывающие переговоры по скайпу, — отнюдь не утопическая фантазия в стиле «1984», а суровая реальность

Через тридцать лет после Оруэлла

В данной статье не рассматриваются этические проблемы ведения наблюдения за пользователями и вопросы, связанные с ними. В данной статье рассматриваются сугубо технические аспекты данных программ, как бы это ни звучало. Любой инструмент, начиная с ножа, можно использовать как во благо, так и во вред.

Каждый работодатель хочет знать, чем сейчас заняты его сотрудники. Пока техника наращивала свои возможности, для этого применялись ухищрения другого рода — например, планирование офисов Open Space. Современные технические решения позволяют это делать гораздо более удобным способом.

Концепция такая — на компьютер пользователя, за которым ведется наблюдение, устанавливается специальный агент, который и передает все необходимые данные на сервер, где через программу просмотра оператор просматривает данные. Операторов может быть несколько, с разными привилегиями, один из них является администратором системы. Обычно присутствует и программа администрирования, которая позволяет установить или удалить агента.

В обзоре будут участвовать следующие программы:

- > StaffCop Standart 5.6

- > LanAgent Enterprise 4.5.6

- > Стах@новец 4.70

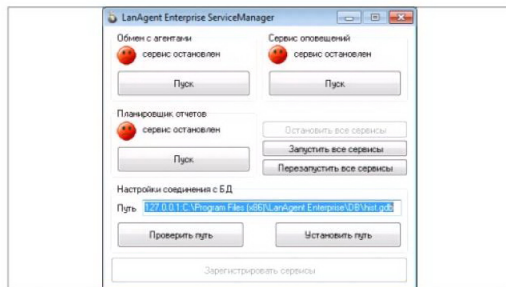
Каждая из них предлагает стандартную для программ такого класса функциональность:

- > информация о запускаемых, устанавливаемых и удаляемых на компьютере программах;
- > запись текста, вводимого с клавиатуры;
- > информация о посещаемых веб-страницах;
- > снимки экрана пользователя;
- > отслеживание коммуникаций по ICQ и Skype;
- > отслеживание включения и выключения компьютера;
- > запись текста, поступающего в буфер обмена;
- > контроль электронной почты;
- > контроль сменных носителей;
- > контроль документов, выводимых на печать;
- > активное оповещение о вводе неразрешенного текста.

Как видите, перечень довольно большой. Поэтому рассматриваться будет не только техническая сторона программы, но и организация системы использования полученных данных.

Что такое «активное оповещение»? Когда пользователь набирает на клавиатуре заданный текст, например, «директор», или когда у него в окне отображается заданный текст, например, «зарплата», оператору приходит уведомление об этом.

Рисунок 1. Интерфейс LanAgent Server



StaffCop

Программа StaffCop разрабатывается новосибирской компанией «Атом Безопасность» [1]. Существуют версии Standart и Home — последняя отличается от Standart тем, что рассчитана на наблюдение за одним компьютером. Стоимость лицензии на Home-версию составляет 990 руб., на Standart в зависимости от числа наблюдаемых компьютеров — от 850 до 1300 руб.

Программа устанавливается на компьютер администратора, на нем же разворачивается база данных. Какого-либо разделяемого доступа к базе не предусмотрено. Все, администрирование и просмотр отчетов, делается из одной программы, поэтому разделить права на просмотр различных

наблюдаемых пользователей невозможно, хотя все типовые функции программ наблюдения представлены полностью.

К сожалению, StaffCop не предлагает Enterprise-версию, рассчитанную на одновременную работу нескольких операторов – предполагается, что в сети есть единственный оператор. Такая концепция характерна для небольших компаний, но сложность (для разработчиков) состоит в том, что обычно компании такого объема не нуждаются в специализированном софте для наблюдения за пользователями. Поскольку тестирование проводилось для внедрения в достаточно большой сети, StaffCop, несмотря на достаточно интересную реализацию, тестировался по сокращенной программе.

LanAgent

Программа LanAgent разрабатывается компанией «Нетворк Профи» [2] и ориентируется на достаточно крупные сети – присутствуют версия Standart, версия Enterprise, специальная версия для терминальных серверов и отдельный модуль LanAgent NetworkFilter. Последний, правда, могут использовать только те, кто доверил шлюз в Интернете операционной системе Windows.

Версия Standart похожа на StaffCop – монолитный модуль, который одновременно и сервер, и консоль администратора, и программа для просмотра отчетов. В относительно крупной сети это неудобно, поэтому тестировалась Enterprise-версия, которая имеет отдельные сервер (LanAgent Server), консоль администратора (LanAgent Administrator) и программу для просмотра отчетов (LanAgent Viewer). Как и должно быть в программах такого класса, Viewer непосредственно не собирает данные – этим занимается сервер. Viewer только подключается к серверу и отображает накопленные данные.

LanAgent использует БД Firebird, которую необходимо установить до установки самого сервера LanAgent. Недо-

работкой, на которую мной уже было указано, является то, что в настройках LanAgent Server путь к БД нужно указывать вручную. Зато это позволяет держать БД на отдельном компьютере. Внешне LanAgent Server – скромная панелька, которая позволяет только проверить состояние и перезапустить в случае необходимости свои сервисы да указать путь к БД (см. рис. 1).

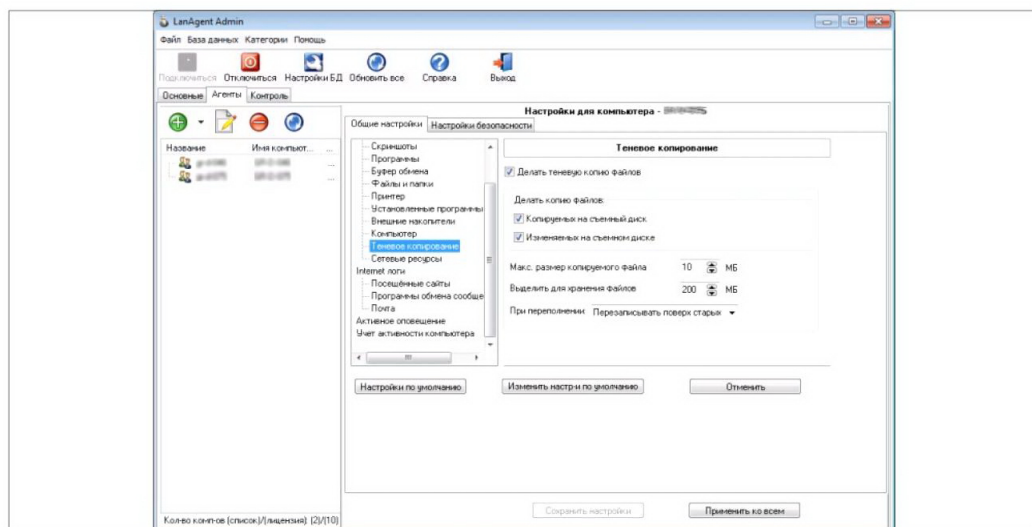
LanAgent Admin – это основная программа для администратора. Именно здесь осуществляются установка и удаление агентов наблюдения на компьютеры пользователей, а также настройка параметров наблюдения. Каждый компьютер имеет свой набор параметров, которые копируются из параметров по умолчанию (которые тоже можно править). Настроек достаточно много, они сгруппированы по вкладкам (см. рис. 2).

Все настройки перечислить трудно, для каждого модуля существует собственный набор. Чтобы получить представление обо всем заявляемом функционале LanAgent, достаточно его просмотреть – тут и скриншоты, и перехват клавиатурного ввода, и контроль активности программ, и теневое копирование файлов, выводимых на съемные носители, и многое другое. Почему я говорю «заявляемый функционал»? Потому что часть его при тестировании оказалась неработоспособной, а еще часть функционировала не так, как ожидалось. Об этом будет рассказано в конце отчета по LanAgent.

Активное оповещение настраивается на соседней вкладке – там можно ввести либо имя программы, либо текст. Срабатывает оно либо при запуске указанной программы, либо при вводе/отображении указанного текста.

Отдельно следует рассказать об установке агентов наблюдения, их удалении и проверке состояния. Почему-то в LanAgent это можно сделать только из диалога добавления пользователя, выбрав сканирование диапазона. Тогда появляется окно, показанное на рис. 3, где можно установить

Рисунок 2. Интерфейс основного окна LanAgent Admin



или удалить агента, а также добавить или удалить компьютер из списка мониторинга. Это две разные операции: первая устанавливает на компьютер определенные файлы и запускает определенные службы, вторая добавляет компьютер в БД. Имейте в виду, что, когда компьютер удаляется из списка мониторинга, все его данные в БД также удаляются!

LanAgent Viewer – это последняя программа комплекса, предназначенная для просмотра данных мониторинга. Организована довольно логично: справа – список компьютеров, слева распределены по закладкам группы данных, которые можно просмотреть. Скриншоты показываются в виде небольшого preview, для просмотра в полном размере следует щелкнуть по нему два раза (см. рис. 4).

Значок в списке возле имени компьютера обозначает текущее состояние передачи данных. Передача данных может быть остановлена по разным причинам, в том числе и из-за противодействия пользователя, а оно, что бы там на сайте [2] ни писали, возможно.

Новые данные на закладках отмечаются соответствующим значком. Данные обновляются автоматически (если это задано в настройках), но тем не менее при установке со значениями по умолчанию проходит порядка семи минут, прежде чем действие, совершенное пользователем, отобразится в программе.

Если компьютер по какой-либо причине вдруг перестал передавать данные, то заново запустить процесс обмена можно, выбрав пункт меню «Формирование задач» (с моей точки зрения, название крайне неудачное). В этом пункте можно обновить логи, остановить или запустить агента, сделать скриншот или отправить сообщение (см. рис. 5).

Отдельно следует также отметить плохую переносимость LanAgent антивирусами. На сайте [2] есть даже целая инструкция, как настраивать исключения для Антивируса Касперского, защитника Windows и некоторых других антиви-

русов. Поэтому, если данные с компьютера периодически перестают идти, проверьте список исключений антивируса – вполне возможно, он просто удалил часть LanAgent.

И, наконец-то, об ошибках функционирования. Тестирование LanAgent проводилось довольно долго, потому что программа обладает приятным интерфейсом, и мне хотелось ее использовать. Но в процессе тестирования обнаружились некоторые неприятнейшие особенности:

- > полностью не работает модуль контроля печати;
- > контроль за посещенными сайтами периодически перестает работать и не возобновляет работу вплоть до перезапуска браузера;
- > при переходе ввода в качестве заголовка окна, в котором вводился текст, подставляется совершенно произвольный заголовок из числа открытых в данный момент;
- > модули перехвата сообщений ICQ и веб-почты не работают;
- > модуль перехвата переговоров по Skype вполне способен вместе с новым разговором достать откуда-то еще и кучу старых;
- > модули перехвата почты и контроля социальных сетей не работают.

Итог: программа с большим функционалом и удобным интерфейсом, но в значительной степени «сырая», несмотря на достаточно почтенный возраст, имеет множество недоработок, причем очень существенных, имеет проблемы с антивирусом и достаточно легко нейтрализуется более-менее грамотными пользователями (в Интернете полно советов о противодействии LanAgent – от весьма глупых до вполне действенных).

Стоимость – от 2100 руб. за один компьютер. При этом следует учесть тот факт, что Enterprise-версия лицензируется не менее чем на 21 компьютер.

Стах@новец

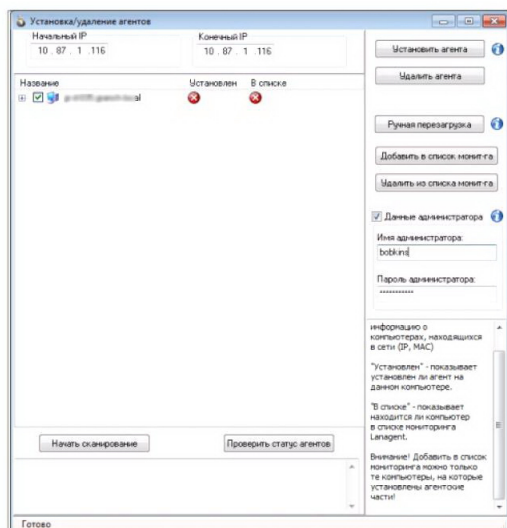
Именно так, через @, пишется его название. Программный комплекс Стах@новец разрабатывается украинской компанией «Бизнес-Лаборатория» [3]. Продажи в России осуществляет Aflex Distribution [4]. Существует также специализированный сайт, посвященный комплексу [5].

При его разработке явно были учтены многие ошибки и технологические недоработки прочих программ. Это видно хотя бы по тому, что в настоящий момент ПК Стах@новец не обнаруживается ни Антивирусом Касперского, ни защитником Windows. И, даже имея перед глазами список запущенных служб Windows, не так-то просто найти среди них агента наблюдения Стах@новца – авторами применен весьма грамотный подход.

У Стах@новца имеется только Enterprise-версия, которая, как и положено, разбита на три компонента: Стах@новец Администратор, Стах@новец Сервер и... все! Клиента нет – вместо него обычный браузер. Таким образом, устраняется необходимость разработки отдельного клиента и достигается кроссплатформенность. Правда, это же порождает страшнейший корявый интерфейс – пожалуй, самый большой недостаток Стах@новца.

Огромным достоинством Стах@новца является то, что он поддерживает как минимум два типа БД – MS SQL и MySQL. Правда, документация не учитывает тот факт, что MySQL может стоять на машине, где ОС – вовсе

Рисунок 3. Интерфейс добавления агента LanAgent Admin



не Windows, но это мелочи. Устанавливать нужно в следующей последовательности:

- > База данных
- > Стах@новец Администратор
- > Стах@новец Сервер

Именно так, а не иначе. Дело в том, что создание структуры БД и загрузка начальных значений в таблицы делается Администратором (программа «Конфигурация базы данных», см. рис. 6). На этапе установки Сервера, БД уже должна быть проинициализирована, иначе можно увидеть ошибку «Не найден пользователь stkh_int_user».

После задания конфигурации БД и ее инициализации запускается программа Сервера «Настройки сервера». Именно в этом месте вводится программный ключ (его надо получить заранее – без ключа будет доступен мониторинг только одного компьютера), указывается расположение сервера БД и ее тип (см. рис. 7).

После записи настроек сервера можно запускать программу «Глобальные настройки», в которой сосредоточены все административные функции. Для запуска потребуется пароль, заданный на этапе настроек, – не забывайте его!

Два самых нужных раздела из данной программы – это «Пользователи базы», в котором создаются внутренние пользователи-операторы Стах@новец (см. рис. 8). Здесь же им назначаются права – на закладке «Дополнительные запреты» можно указать список сотрудников, которые будут доступны данной учетной записи.

Второй, самый нужный, раздел – это «Настройки комплекса». Настроек здесь много, ничуть не меньше, чем в LanAgent, правда, сгруппированы они по-другому – есть настройки сервера («Для сервера»), есть настройки пользовательских компьютеров («Для компьютеров»), есть настройки собственно пользователей («Для пользователей»). В отличие от LanAgent здесь на всех пользователей изначально

Рисунок 4. Интерфейс основного окна LanAgent Viewer

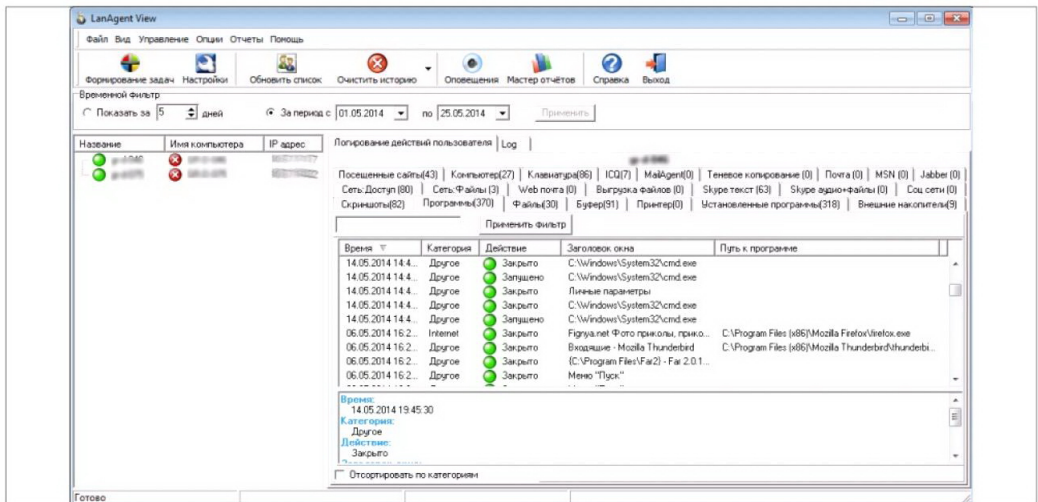
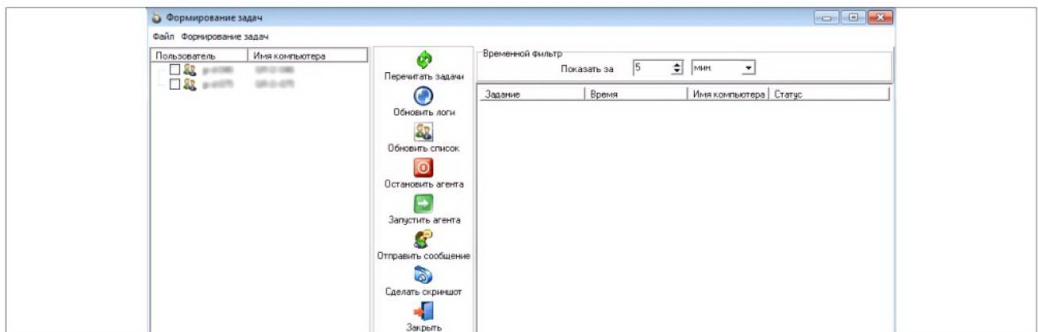


Рисунок 5. Интерфейс подменю «Формирование задач» LanAgent Viewer



распространяется один набор настроек. Чтобы создать отдельный набор настроек, нужно создать группу и изменить в ней настройки, требующие изменения.

Ограниченный объем журнальной статьи не позволяет мне даже перечислить тут все настройки, не то, что описать их. Их много. Можно вести постоянную аудиозапись с микрофона компьютера, можно выполнять и фиксировать снимки через веб-камеры, блокировать запись на съемные носители, запрещать передачу вложенных файлов по почте, в том числе и по веб-почте, – для ознакомления с полным списком возможностей лучше все попробовать самим (см. рис. 9).

Еще в «Глобальных настройках» есть «Досье сотрудников», где можно указать для каждого пользователя его фамилию, занести фотографию и «Журнал», где отображается, кто и когда заходил в систему и каким модулем пользовался. Остальные пункты предназначены для обслуживания и не интересны.

Ну и последнее – это модуль просмотра отчетов. Как такового клиента нет, его заменяет устанавливаемый на компьютер, где запущен Стах@новец Сервер, веб-сервер Apache, который и обеспечивает доступ к отчетам.

Для доступа к отчетам предложены два модуля – БОСС-Онлайн и БОСС-Оффлайн. Для подключения к любому из них необходимо указать пароль той самой учетной записи, которая создавалась в программе Администратора. БОСС-Онлайн – это оператор с функциями администратора, например, только в БОСС-Онлайн возможно удалить установленного агента.

Но основная задача БОСС-Онлайн (см. рис. 10) – оперативное наблюдение. В левой части экрана отображается та задача, которая выполняется на наблюдаемом компьютере в данный момент. Компьютеры располагаются в виде списка, таким образом, можно наблюдать, чем заняты все пользователи одновременно, – такого у LanAgent нет. В зависимости от того, на какую строку установлен курсор – на компьютер или на подключенного к этому компьютеру пользователя, в правой верхней части доступны различные инструменты. Для компьютера возможна работа с веб-камерами и микрофонами, обновление или удаление агента, а для пользователя – немедленный скриншот или непрерывное наблюдение за экраном (работает только в Internet Explorer) или отправка сообщения.

Рисунок 6. ПК Стах@новец. Инициализация базы данных

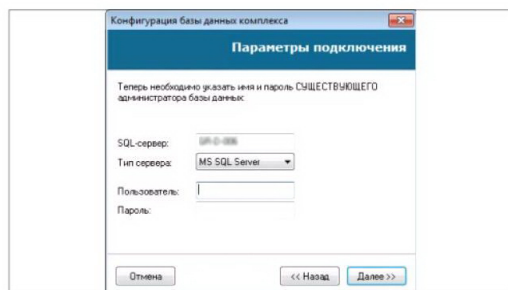


Рисунок 7. ПК Стах@новец. Настройка сервера

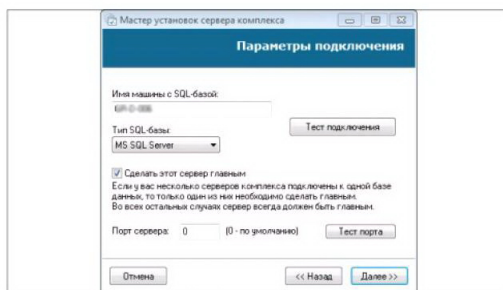
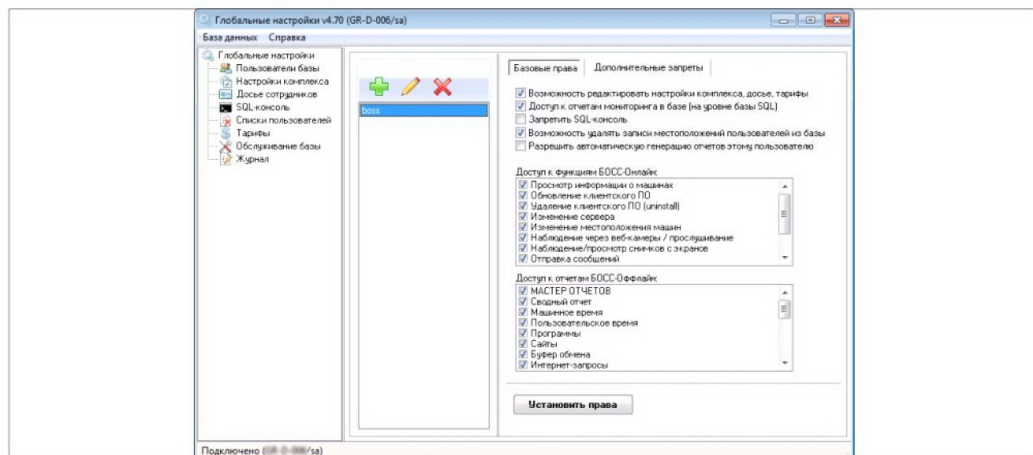


Рисунок 8. ПК Стах@новец. Создание пользователя-оператора



В правой нижней части выводится список оповещений по ключевым словам, задаваемым в программе Глобальных настроек – такое здесь активное оповещение. Активное оповещение можно сделать на почту или по FTP, на самом деле продумано достаточно много. Но основная работа все же происходит в модуле БОСС-Оффлайн, который отображает отчеты, сформированные по данным, накопленным в БД. Единственное, что портит вид, – это страшненький интерфейс – ну HTML же.

Здесь в полной мере проявляется то преимущество, которого нет у LanAgent, – возможность построения групповых отчетов, в особенности группового сводного отчета по всем сотрудникам (см. рис. 11). В этом отчете отображается все рабочее время сотрудника от входа в систему до выхода, причем считаются все входы-выходы в течение дня, время, проведенное им в программах, и время, проведенное в Интернете, и здесь же отображаются наиболее популярные для него в этот день программы и интернет-сайты. Для аналитика это самый подходящий отчет, когда нужно быстро оценить, кто работал, а кто сидел в Интернете и разглядывал сомнительные картинки.

Остальные отчеты позволяют выяснить, чем занимался пользователь более детально. Отчет «Пользовательское время» (см. рис. 12) показывает распределение времени в течение дня – когда человек работал, а когда отсутствовал за компьютером.

Отчет «Программы» отображает все запущенные за отчетный период программы, причем они отсортированы по частоте использования. Если в программе вводился текст, то внизу появится пункт «Показать/скрыть вводимый текст».

К сожалению, перехватчик текста довольно часто сбоит, и текст отображается не в порядке ввода, а в каком-то совершенно произвольном порядке и превращается в совершенно нечитаемую чушь.

Отчет «Сайты» отображает все посещенные сайты, причем точно так же они будут отсортированы по частоте использования, и, если при их посещении вводился какой-либо текст, ниже пункта по данному сайту появится строчка

«Показать/скрыть вводимый текст». По умолчанию сайты группируются, но можно группировку отменить.

Отчет «Буфер обмена» показывает содержимое буфера обмена, отмечая время, когда в него поступил данный фрагмент.

Отчет «Интернет-запросы» показывает вводимые поисковые запросы. Гарантированно отображаются только поисковые запросы в Google, остальные можно перехватить в отчетах «Программы» или по снимкам.

Отчет «Снимки экранов» показывает снимки экрана пользователя. Причем делаются они и в то время, когда пользователь заблокировал компьютер и ушел. Среди множества настроек у Стах@новец отсутствует настройка не делать скриншоты, когда компьютер заблокирован или выключен.

Отчет «Печать на принтере» (см. рис. 13) отображает список файлов, выведенных на печать. Копии распечатанных файлов сохраняются. Отображается время, объем файла и количество страниц.

Отчет «Файловые операции» отображает операции с файлами. К сожалению, работает этот отчет только для файлов на диске С, хотя иногда срабатывает и для других дисков и отслеживает только некоторые файловые операции, поэтому он относительно бесполезен.

Отчет «Отправка файлов» отображает все файлы, переданные в качестве вложений в сообщениях электронной почты.

Отчет «Почта» отображает всю переписку пользователя за отчетный период. Отображаются все входящие и исходящие письма. Копии писем сохраняются вместе с вложениями. Отчет работает как с почтой IMAP/POP3, так и с MAPI (MS Exchange), в отличие от LanAgent, где переписка по MAPI не контролируется.

Отчет «Чаты/звонки» отображает переписку по ICQ и Skype.

Отчет «События: пользователь» отображает события в деятельности пользователя – печать на принтере, вывод файлов на сменный носитель, отправка файлов по Интернету, ввод слова, на которое настроено активное оповещение.

Рисунок 9. ПК Стах@новец. Настройки пользователя

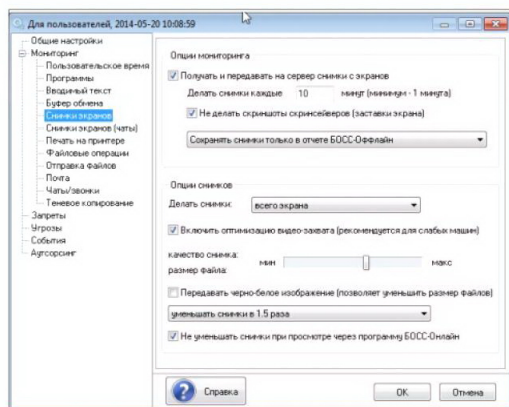
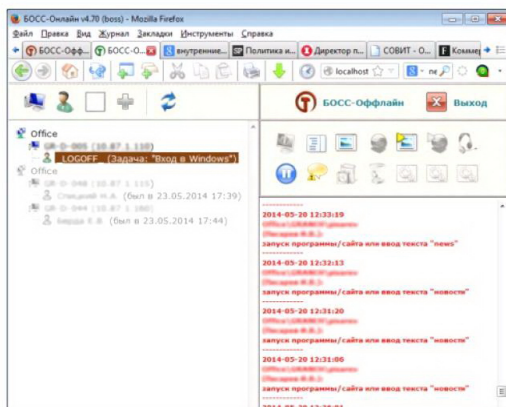


Рисунок 10. ПК Стах@новец. Основной экран БОСС-Онлайн



Отчет «Оборудование/софт» отображает информацию об аппаратной и программной конфигурации компьютера. Отображается она в страшно неудобном виде – чем-то напоминает отладку скрипта на PHP.

С моей точки зрения, лучшей программой является Стах@новец. Да, у него не лучший интерфейс. Зато он лишен тех проблем, что зачастую просто не дают работать LanAgent, – беспричинное пропадание связи с наблюдаемым компьютером, что приводило к необходимости постоянно переставлять агента. Ну и, кроме того, групповые отчеты Стах@новеца позволяют дать быструю оценку ситуации, у LanAgent же

потребуется запуск мастера построения отчетов. Пользователь, помни: Большой Брат видит тебя! **ЕОБ**

1. Сайт программы StaffCop – www.staffcop.ru.
2. Сайт программы LanAgent – <http://www.lanagent.ru>.
3. Сайт разработчика программы Стах@новец – <http://probusinesslab.com>.
4. Представитель разработчика программы Стах@новец в России – <http://www.aflex.ru>.
5. Сайт программы Стах@новец – <http://stakhanovets.ru>.

Ключевые слова: контроль работы сотрудников, анализ использования ПК, мониторинг.

Рисунок 11. ПК Стах@новец. Сводный отчет по сотрудникам

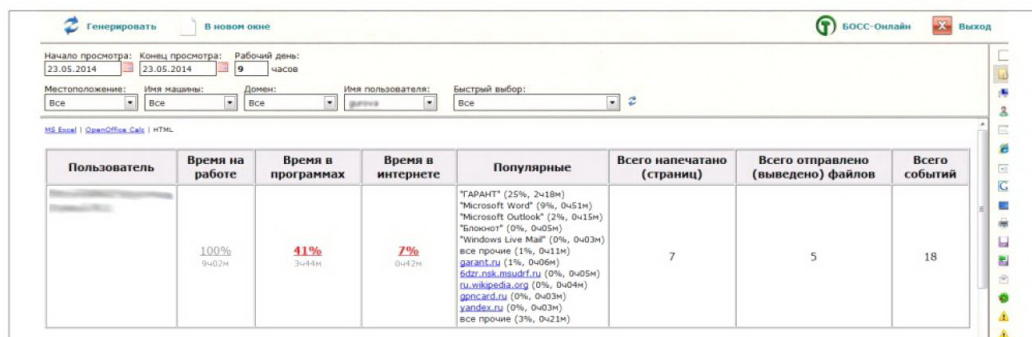


Рисунок 12. ПК Стах@новец. Отчет «Пользовательское время»

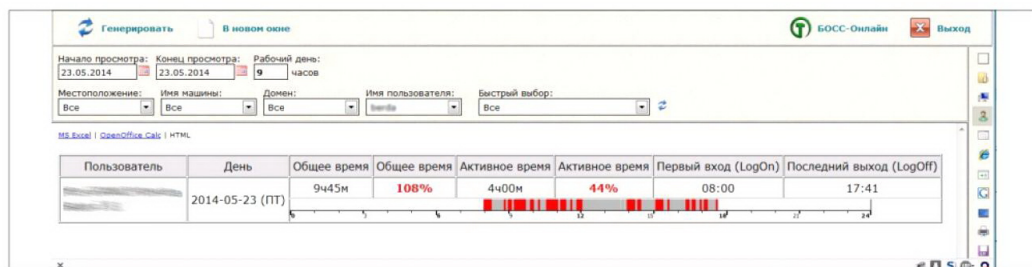
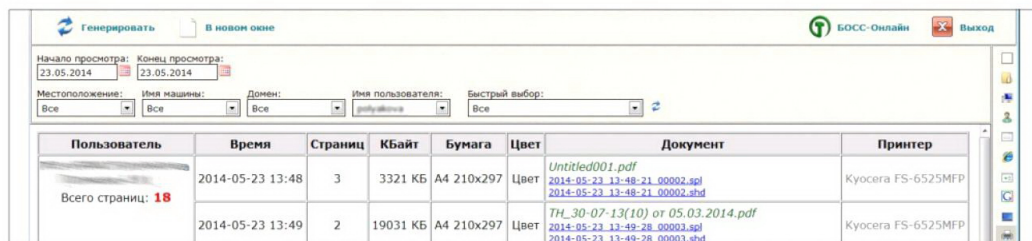


Рисунок 13. ПК Стах@новец. Отчет «Печать на принтере»





Как избежать ошибок учета в «1С»

Часть 11. Налог на добавленную стоимость

Учет налога на добавленную стоимость регламентируется законодательными документами. Поэтому важно обеспечивать корректность операций

Вам стоимость добавить?

Как и в ранее опубликованных статьях [1-11, 14], договоримся, что приведенные примеры используют конфигурацию «Управление производственным предприятием 1.3» на базе платформы «1С:Предприятие 8.2» (далее по тексту УПП). Кроме того, включен режим расширенной аналитики учета запасов и затрат (РАУЗ), преимуществами и недостатки которого были подробно описаны в [12-13].

Налог на добавленную стоимость (далее по тексту НДС) – это особый косвенный налог, который позволяет изъять в бюджет государства часть стоимости товара, работы или услуги, появляющуюся на каждом из этапов прироста ее стоимости. НДС вносится в бюджет по мере реализации.

Если смотреть на механизм в целом, становится очевидным, что налог со всей стоимости приобретаемого товара, работы или услуги в итоге уплачивает продавцу конечный потребитель. Однако в бюджет эта сумма начинает поступать раньше, чем осуществлена последняя реализация.

Каждый, кто участвует в производстве товара, работы или услуги, на разных стадиях ее готовности уплачивает на-

лог с той части стоимости, которая добавляется к стоимости приобретенного сырья, работ и услуг, необходимых для выпуска на данном производственном этапе.

Звучит достаточно сложно и запутанно, но смысл относительно прост. Если предприятие закупает для производства продукции материалы, работы и услуги, а затем продает результат труда с наценкой, то последняя и есть та самая пресловутая «добавленная стоимость», которая является объектом налогообложения.

В России налоговая ставка НДС составляет 18%. Она применяется в большинстве случаев, за исключением операций из перечня облагаемых по ставке 10% или перечня со ставкой 0%.

Учет НДС включает в себя три этапа:

- > выделение НДС при покупке товаров или услуг;
- > начисление НДС при продаже товаров или услуг;
- > принятие НДС к вычету.

Параметры учета НДС для организации настраиваются в учетной политике на закладке «НДС». При этом, если, например, организация использует упрощенную систему налогообложения, то есть не является плательщиком НДС,

Рисунок 1. Для каждой товарной позиции в документе «Поступление товаров и услуг» нужно указывать ставку НДС

№	Номенклатура	Серия	Количес.	Ед.	К.	Цена	Сумма	% НДС	Сумма НДС	Всего	Заказ
1	Материал №1		1 000 000	шт	1,000	560.00	560 000.00	18%	85 423.73	560 000.00	
2	Материал №2		1 200 000	шт	1,000	1 800.00	2 160 000.00	0%		2 160 000.00	

Тип цен: Не заполнено!
 Счет-фактура: № 403 от 01.04.2014 г. (Счет-фактура получена от 000000000001 от 01.04.2014 г.)
 Комментарий:
 Всего (руб.): 2 720 000.00
 НДС (в т. ч.): 85 423.73

ТОРГ-12 (Товарная накладная за поставками с услугами) Печать: ОК | Записать | Закрыть

раздел «НДС» в форме учетной политики просто не отображается.

На закладке «Налог на добавленную стоимость (НДС)» выбирается порядок расчетов с покупателями товаров при частичной оплате документа, в котором присутствуют разные ставки НДС.

- > Если заранее известно, что для части товаров из документа НДС невозможно принять к вычету, а другие, наоборот, будут приняты к вычету со 100% вероятностью, то можно указать, какие из них признаются оплаченными в первую очередь.
- > Если в документе перечислены товары, облагаемые НДС по различным ставкам, то можно задать порядок оплаты, например, так, что первыми считаются оплаченными товары по ставке 0% или же они, наоборот, идут в последнюю очередь.
- > Порядок формирования счетов-фактур и учета НДС по суммовым разницам указывается на закладке «Расчеты в условных единицах (у.е.)».

Для того чтобы суммовые разницы учитывались как суммы, связанные с расчетами по оплате согласно ста-

тье 162 НК РФ, необходимо установить флаг «Формировать счета-фактуры на суммовые разницы». Возможен учет только положительных суммовых разниц.

Непосредственно начисление НДС по суммовой разнице производится в момент проведения документов оплаты. Для правильного отражения в учете по начисленным суммам нужно сформировать счета-фактуры с помощью обработки «Регистрация счетов-фактур на суммовые разницы». В последующем суммы начисления отражаются в строках 160-170 раздела 3 декларации по НДС.

Если организация осуществляет реализацию с НДС 0% или без НДС, то на закладке «Реализация без НДС или с НДС 0%» необходимо установить флаг «Организация осуществляет реализацию без НДС или с НДС 0%». После установки флага для организации будет доступен учет партий в целях контроля состояния НДС, предъявленного поставщиками товаров. Прежде всего это означает включение НДС в стоимость материально-производственного запаса и в состав расходов при реализации без НДС или с НДС 0% и принятие НДС к вычету после подтверждения нулевой ставки.

Рисунок 2. Важным условием корректности учета НДС является оформление счета-фактуры на поступление

Рисунок 3. Как и при поступлении, в документе «Реализация товаров и услуг» ставка НДС указывается для каждой товарной позиции

№	Номенклатура	Серия	Количество	Ед.	К.	Су.	Цена	% НДС	Сумма НДС	Всего	Способ оплаты
1	Материал NF1		1 000.000	шт	1.000	7...	700.00	18%	106 779.66	700 000.00	Со склада
2	Материал NF2		1 200.000	шт	1.000	1...	1 500.00	0%	1 800 000.00	1 800 000.00	Со склада

Выделение, начисление и принятие

При оформлении поступления товаров или услуг есть возможность указать, по какой ставке НДС учитывается каждая из позиций. Например, на рис. 1 товарная позиция «Материал №1» поступает с НДС 18% (сумма НДС равна 85 423,73 руб.), а «Материал №2» – с НДС 0%.

Очень важно, что для корректного учета НДС, кроме документа поступления, должен быть оформлен соответствующий счет-фактура (см. рис. 2).

В соответствии с законодательством (глава 21 НК РФ) существует ряд событий, которые приводят к обязанности налогоплательщика начислить НДС к уплате в бюджет.

Основным событием, при котором возникает обязанность начисления НДС, является реализация товаров, работ и услуг на территории РФ (статья 146 НК РФ).

В налоговую базу также включаются прочие суммы, поступившие в счет реализации товаров, работ или услуг, например, авансовые платежи в счет будущих поставок.

Кроме того, особый порядок предусмотрен для отражения операций со ставкой НДС 0% (глава 21 НК РФ). Выручка по таким сделкам попадает в декларацию только после подтверждения или неподтверждения права применения нулевой процентной ставки.

Наконец, есть специфические случаи начисления НДС: исполнение обязанностей налогового агента, выполнение строительно-монтажных работ для собственного потребления внутри организации. В то же время основными документами, при проведении которых формируются бухгалтерские проводки по дебету счета учета НДС по реализации и кредиту счета 68.02 «Налог на добавленную стоимость», являются:

- > реализация товаров и услуг;
- > реализация услуг по переработке;
- > отчет комиссионера о продажах;
- > отчет комитенту о продажах товаров;

- > отчет о розничных продажах;
- > передача ОС;
- > передача НМА.

При заполнении документа «Реализация товаров и услуг», так же как и при оформлении поступления, в табличной части указывается ставка НДС для каждой продаваемой товарной единицы (см. рис. 3). Для корректности учета НДС на реализацию также вводится счет-фактура (см. рис. 4).

Чудо-книги

Для целей корректного ведения учета НДС в УПП предусмотрены два документа: «Формирование записей книги покупок» и «Формирование записей книги продаж».

Документ «Формирование записей книги покупок» предназначен для регистрации вычетов сумм НДС (см. рис. 5, сверху).

Табличные части документа доступны для редактирования. У документа есть два режима заполнения:

- > принятие НДС к вычету по ценностям, используемым для «обычных» операций, облагаемых НДС по любой из ставок, кроме 0%;
- > принятие НДС к вычету по ценностям, использованным для операций, облагаемых НДС по ставке 0%.

Выбор режима в каждом конкретном документе осуществляется с помощью флага «Предъявлен НДС к вычету 0%». Табличная часть «Вычет НДС по приобретенным ценностям» позволяет регистрировать в книге покупок суммы НДС, уплаченные поставщикам приобретенных ценностей.

Для принятия к вычету «обычных» операций проверяется выполнение следующих обязательных условий:

- > наличие счета-фактуры (только для НДС от поставщиков);
- > оплата (только для НДС от поставщиков);
- > предъявление суммы НДС поставщиком (или уплата НДС на таможене);

Рисунок 4. Счета-фактуры нужно оформлять не только на поступления, но и на реализации товаров и услуг

Счет-фактура выданный: Проведен

Действия: [Иконки]

Номер: 0000000000001 от: 05.05.2014 12:00:00

Организация: ООО "Наша фирма"

Контрагент: ООО "Наш поставщик"

Договор: Договор

Вид счета-фактуры: На реализацию

Исправление номер: 0

Основное | Дополнительно

Документ-основание счета-фактуры

№ Документ - основание для выписки счета-фактуры

Реализация товаров и услуг 0000000000001 от 05.05.2014 12:00:00

Реализация не подлежит налогообложению (освобождена от налогообложения)

Платежно-расчетный документ

Дата плате... Номер платежно-расчетного документа

Выставление счета-фактуры

Код вида операции: 01

Не выставляется

Выставлен: 05.05.2014

На бумажном носителе | В электронном виде

Состояние 3Д: Обмен электронными документами отключен

Всего: 2 500 000.00 руб. НДС (е.т.ч.): 106 779.66

Счет-фактура | Печать | ОК | Записать | Закрыть

- > отсутствие включения НДС в стоимость;
- > отсутствие связи с применением ставки НДС 0%;
- > отсутствие уже состоявшегося факта принятия НДС к вычету.

Для основных средств и строительно-монтажных работ проверяется, введены ли они в эксплуатацию. Для принятия к вычету по операциям с НДС 0% проверяется:

- > наличие счета-фактуры (только для НДС от поставщиков);
- > оплата (только для НДС от поставщиков);
- > предъявление суммы НДС поставщиком (или уплата НДС на таможене);
- > подтверждение или неподтверждение права применения ставки 0%;
- > отсутствие состоявшегося факта принятия НДС 0%.

При проведении документа формируются записи по регистрам учета НДС.

В бухгалтерском учете формируются записи по кредиту счета 19 «Налог на добавленную стоимость по приобретенным ценностям» и по дебету счета 68.02 «Налог на добавленную стоимость». Конкретные субсчета определяются видом приобретенной ценности по данным регистра «НДС покупки». Табличная часть «Вычет НДС с авансов» предназначена для автоматизированного предъявления к вычету суммы налога на добавленную стоимость, ранее исчисленных с полученных авансов.

Корректные результаты могут быть получены только после регистрации в системе налогового учета счетов-фактур на полученные авансы.

При проведении документа формируются записи по регистрам учета НДС и бухгалтерские проводки по кредиту счета 76.АВ «НДС по авансам и предоплатам» и дебету счета 68.02 «Налог на добавленную стоимость».

Документ «Формирование записей книги продаж» предназначен для регистрации сумм НДС, которые напрямую связаны с начислением НДС к уплате в бюджет (см. рис. 5, снизу).

У документа существует два альтернативных режима:

- > НДС по реализации, облагаемой по обычным ставкам НДС (18%, 10% и т.п., кроме ставки 0%);
- > НДС по реализации по ставке 0%.

При переключении режима считается, что начато заполнение нового документа, и все заполненные данные очищаются. Табличные части документа могут быть заполнены автоматически по кнопке «Заполнить», однако они доступны и для редактирования вручную.

При проведении документа формируются записи в регистрах учета НДС.

Табличная часть «НДС по реализации» позволяет регистрировать в книге продаж суммы НДС, начисленные при реализации ценностей.

При автозаполнении в табличную часть попадают данные по реализации с различными ставками НДС. При этом проверяется наличие оплаты по счету-фактуре, и в табличную часть попадает только оплаченная сумма (для политики налогового учета в части НДС «по оплате»). Если оплата по счету-фактуре была частичной, то в документе появятся две строки – сумма оплаты с указанием документа и остаток (при политике «по отгрузке»).

В отчет «Книга продаж» они попадут одной строкой.

При установленном флаге «По реализации со ставкой 0%» в табличную часть попадают только строки реализации со ставкой 0%, по которым был сформирован документ «Подтверждение нулевой ставки НДС».

Кроме записей в регистрах учета НДС, формируются бухгалтерские проводки по кредиту счета 68.02 «Налог

Рисунок 5. При корректном ведении учета книги покупок и продаж формируются автоматически по кнопке «Заполнить» на основании внесенных документов

Формирование записей книги покупок: Проведен

Номер: 000000000001 от: 17.05.2014 14:34:02

Организация: ООО "Наша фирма" Ответственный:

☐ Предъявлен к вычету НДС 0%

Вычет НДС по приобретенным... Вычет НДС с полученных авансов... Вычет НДС с выданных авансов... Вычет НДС по налоговому акту... Вычет НДС по уменьшению стоимости...

№	Вид ценности	Поставщик	Счет-фактура	Сумма без НДС	% НДС	НДС	Д	Дата оплаты	Счет учета
1	Материалы	ООО "Наш постав..."	Поступление товаров и услуг 0...	474 576,27	18%	85 423,73		19.03	
2	Материалы	ООО "Наш постав..."	Поступление товаров и услуг 0...	2 160 000,00	0%			19.03	

Комментарий:

Печать OK Записать Закрыть

Формирование записей книги продаж: Проведен

Номер: 000000000001 от: 17.05.2014 14:47:17

Организация: ООО "Наша фирма" Ответственный: ИР

☐ По реализации со ставкой НДС 0%

По реализации (2 пос.) С авансов (0 пос.) Начислен к уплате (0 пос.) Возврат по авансам (0 пос.) Возврат по другим операциям (0 пос.) Не отражается в книге (0 пос.)

№	Вид ценности	Покупатель	Счет-фактура	% НДС	Сумма без НДС	НДС	Документ оплаты	Дата оплаты	Соб.	Запись деп.	К...
1	Материалы	ООО "Наш пос..."	Реализация товаров...	18%	169 491,53	30 508,47	Платежное поручение в...	20.01.2014	Рез.		
2	Материалы	ООО "Наш пос..."	Реализация товаров...	18%	423 728,81	76 271,19	Платежное поручение в...	24.02.2014	Рез.		

Комментарий:

Печать OK Записать Закрыть

на добавленную стоимость» и по дебету счета 76.Н «Расчеты по НДС, отложенному для уплаты в бюджет».

Табличная часть «НДС с авансов» позволяет регистрировать в книге продаж суммы НДС по авансам, полученным от покупателей. Она не отображается при выбранном значении «По реализации со ставкой 0%». При автозаполнении в табличную часть попадают данные по авансам, для которых выписаны счета-фактуры.

Табличная часть «НДС начислен к уплате» позволяет регистрировать суммы НДС, начисленные к уплате в бюджет по строительно-монтажным работам, выполненным хозяйством, или по договорам налогового агента.

Табличная часть «Не отражается в книге продаж» предназначена для вывода операций, не требующих отражения в книге продаж.

Разбираем ошибки

Поскольку учет НДС представляет собой многоэтапный процесс, на каждом из шагов возможно возникновение различных ошибок, которые нужно своевременно устранять. Ниже будут разобраны наиболее типичные ошибки при учете НДС.

Не формируются счета-фактуры на аванс

При работе с НДС в УПП очень важно знать, что существуют некоторые регламентные операции для обеспечения корректности учета. В частности, может показаться ошибкой, что автоматически не формируются счета-фактуры на аванс.

Однако на самом деле просто необходимо периодически выполнять обработку «Регистрация счетов-фактур на аванс», которая позволяет сформировать и зарегистрировать в подсистеме налогового учета автоматизированным методом счета-фактуры на авансы, полученные за некоторый определенный период времени (см. рис. 6). Период, за который будут сформированы счета-фактуры, выбирается в соответствующих полях.

Механизм функционирования обработки следующий: сначала определяются остатки авансов на каждую дату периода. Затем в соответствии с порядком регистрации счетов-фактуры на авансы, указанным в учетной политике

или договоре с контрагентом, для каждого из авансов определяется, требуется ли выписать счет-фактуру с учетом даты поступления аванса.

Очень важно, что ставки НДС указываются для каждого из авансов отдельно. Если используется автоматическое заполнение табличной части, по умолчанию для аванса указывается расчетная ставка, соответствующая ставке НДС из платежного документа. При этом можно указать «руками» и любую другую ставку. Сумма НДС в таком случае автоматически пересчитывается исходя из вновь указанной ставки.

Кроме того, если есть такая необходимость, можно ввести разные строки с указанием различных ставок НДС вместо одной строки, которая предлагается автоматически, а также внести любые другие необходимые ручные правки перед проведением.

Запуск обработки стартует по нажатию на кнопку «Выполнить». При этом осуществляются следующие действия:

- > создаются счета-фактуры на аванс (отличительным их признаком является номер, начинающийся с литеры «А»);
- > созданные счета-фактуры автоматически регистрируются в учете (проводятся), в результате формируются:
 - » записи в регистре «НДС начисленный» на сумму НДС, начисленного с аванса;
 - » записи в регистре «НДС с авансов» для целей отслеживания вычетов НДС с аванса.

Очевидно, что может сложиться ситуация, при которой за заданный период ранее уже были сформированы счета-фактуры на аванс. Программа отслеживает такие моменты и предлагает обновить их или повторно использовать для других авансов во избежание двойной регистрации счетов-фактур и сбоя нумерации. Можно также и вовсе отменить формирование этих счетов-фактур на аванс.

На этом сервис отслеживания коллизий при регистрации не заканчивается: в момент формирования счетов-фактур на аванс проверяется возможность их выписки.

Это полезно в случае, когда по указанному документу уже был ранее сформирован счет-фактура на аванс с полностью идентичными реквизитами. Поскольку при выполнении обработки введенные ранее счета-фактуры на аванс

Рисунок 6. Счета-фактуры на аванс регистрируются с помощью специальной обработки

N	Контрагент	Договор	Сумма ав.	Ставка Н.	Сумма НДС	Документ основа	Счет на	Дата	Валюта д.	Валюта	Счет-факт.
1	ООО "Наш поставщик"	Договор	200 000.00	18%	36 000.00	Платежное поруч.	Закас	20.01.2014	руб.	200 000.00	
2	ООО "Наш поставщик"	Договор	565 800.00	18%	101 844.00	Платежное поруч.	Закас	24.02.2014	руб.	565 800.00	
3	ООО "Наш поставщик"	Договор	1 300 000.00	0%		Платежное поруч.	Закас	24.03.2014	руб.	1 300.00	
4	ООО "Наш поставщик"	Договор	412 000.00	18%	74 160.00	Платежное поруч.	Закас	21.04.2014	руб.	412 000.00	
5	ООО "Наш поставщик"	Договор	335 000.00	18%	60 300.00	Платежное поруч.	Закас	10.05.2014	руб.	335 000.00	
6	ООО "Наш поставщик"	Договор	200 000.00	18%	36 000.00	Платежное поруч.	Закас	12.05.2014	руб.	200 000.00	
7	ООО "Наш поставщик"	Договор	200 000.00	18%	36 000.00	Платежное поруч.	Закас	12.05.2014	руб.	200 000.00	
8	ООО "Наш поставщик"	Договор	200 000.00	0%		Платежное поруч.	Закас	12.05.2014	руб.	200 000.00	
9	ООО "Наш поставщик"	Договор	200 000.00	18%	36 000.00	Платежное поруч.	Закас	12.05.2014	руб.	200 000.00	
10	ООО "Наш поставщик"	Договор	200 000.00	18%	36 000.00	Платежное поруч.	Закас	12.05.2014	руб.	200 000.00	

не удаляются, а возможное дублирование отслеживается, то новый счет-фактура создан не будет.

В ходе выполнения обработки в каждой из строк можно видеть признак регистрации конкретного счета-фактуры на аванс.

Увидеть все счета-фактуры на аванс, зарегистрированные по выбранной организации за заданный период можно, нажав кнопку «Список счетов-фактур (выд.)». Данный список полезно видеть как до выполнения обработки регистрации (видно, что уже есть), так и после – для того чтобы отследить, какие документы были сформированы.

Несмотря на то что групповая печать счетов-фактур на аванс в данной обработке не предусмотрена, это всегда можно сделать с помощью универсальной обработки «Групповая обработка справочников и документов», как и для всех остальных документов.

Формируются счета-фактуры не со всех авансов

В ходе работы в программе может оказаться, что формируются счета-фактуры не со всех авансов. При этом регламентная операция запуска обработки «Регистрация счетов-фактур на аванс» выполняется, не выдает ошибок, но результат все равно неправильный.

В этом случае нужно настроить порядок регистрации счетов-фактур на авансы. Механизм здесь следующий: при получении предварительной оплаты от покупателя поставщик обязан выставить счет-фактуру на аванс. При этом согласно законодательству выписывать счета-фактуры на авансы немедленно вовсе не обязательно. Можно это сделать не сразу, а по прошествии некоторого времени с момента получения аванса, в течение которого не наступит исполнение обязательств по договору (отгрузка товаров, оказание услуг).

Описанный вариант выставления счетов-фактур на авансы позволяет избавиться от необходимости формирования документов на платежи, которые закрываются в течение месяца.

Для настройки алгоритма в УПП необходимо войти в интерфейс «Заведующий учетом», выбрать пункт меню «Настройка учета → Учетная политика → НДС» (см. рис. 7). Кроме того, попасть сюда можно прямо из обработки «Ре-

гистрация счетов-фактур на аванс», нажав на соответствующую ссылку с названием текущего значения. Вариант подписи у ссылки зависит от того, какой порядок регистрации выбран на данный момент.

Всего настройка позволяет указать порядок регистрации счетов-фактур на авансы, принятый в организации из следующего списка:

- > регистрировать счета-фактуры на авансы всегда при получении аванса;
- > не регистрировать счета-фактуры на авансы, зачтенные в течение пяти календарных дней;
- > не регистрировать счета-фактуры на авансы, зачтенные до конца месяца;
- > не регистрировать счета-фактуры на авансы, зачтенные до конца налогового периода (Постановление Президиума ВАС РФ от 10.03.2009 №10022/08);
- > не регистрировать счета-фактуры на авансы (п.13 ст.167 НК РФ).

Соответственно, если выбран порядок регистрации авансов, отличный от того, что закреплен в бумажной учетной политике организации, возникнут расхождения в определении, что является авансом, а что нет. Это ведет к прямым ошибкам.

Не попадают записи в книги

Обработка «Проведение документов по регистрам НДС» используется для перепроведения документов по регистрам учета НДС.

При этом задействованы следующие регистры:

- > «НДС, включенный в стоимость»;
- > «НДС по косвенным расходам»;
- > «НДС начисленный»;
- > «НДС по незавершенному производству»;
- > «НДС по партиям запасов»;
- > «НДС предъявленный»;
- > «НДС предъявленный, реализация 0%»;
- > «НДС по реализации 0%»;
- > «НДС с авансов»;
- > «НДС по ОС, НМА».

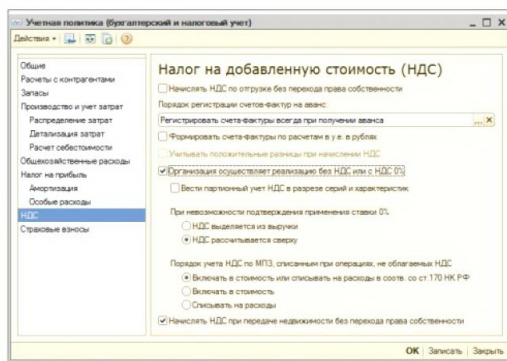
В ходе исполнения обработка совершает следующие действия:

- > проведение документов по оперативным регистрам НДС, аналогичное тому, которое выполняется при проведении самих документов;
- > проведение документов по регистрам партий для целей НДС;
- > отражение суммовых и курсовых разниц для целей НДС.

При выполнении обработки можно выбрать, будет ли осуществлено проведение по всем участкам либо только по части из них.

Чрезвычайно важным является тот факт, что при использовании режима расширенной аналитики учета и затрат запуск обработки является обязательной к выполнению регламентной операцией, если в учете есть операции принятия к учету основных средств, корректировки поступления или реализации, а также при ведении сложного учета НДС (в учетной политике для бухгалтерского и налогового учета установлен флажок «Организация осуществляет реализацию без НДС или с НДС 0%»).

Рисунок 7. Основные параметры учета НДС задаются в учетной политике организации на соответствующей закладке



Если проведение документов по регистрам НДС не выполнить, получится ситуация, когда книги сформированы неправильно и корректность учета нарушена. Все это приводит к ошибкам. Во избежание подобных проблем достаточно выполнить обработку с установленным флагом «Проводить документы по регистрам партий» (см. рис. 8).

Ошибки при ведении сложного учета НДС

Проблематика сложного учета НДС актуальна для производственных предприятий, занимающихся продажей как на территории России (по ставке НДС 18%), так и торгующих на экспорт (ставка НДС 0%). Функционал сложного учета НДС – это, можно сказать, больное место конфигурации УПП в режиме ПАУЗ. Ибо предельно корректно работает он только в случае купли-продажи товаров или материалов. Если же дело касается реализации продукции собственного производства, то сразу возникает ворох очень серьезных проблем, и приходится искать обходные пути.

Более подробно о том, какие ограничения здесь накладывает УПП и как их можно обойти, будет рассказано в следующей статье, посвященной сложному учету НДС.

Для обеспечения корректности учета налога на добавленную стоимость необходимо держать в голове и учитывать множество факторов. Во многом автоматизация с использованием УПП облегчает этот трудоемкий процесс.

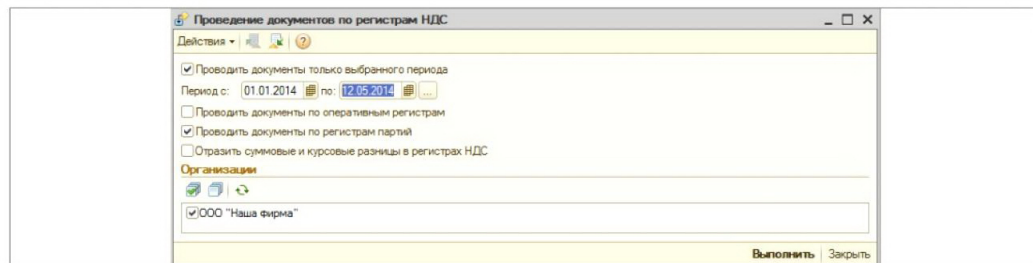
Поскольку нормы ведения учета НДС регламентированы законодательно, большой вариативности нет, но и простое обеспечение правильности учета – серьезная задача сама по себе. **BOX**

1. Чуфаров И. Парадоксы учета в 1С. //«Системный администратор», №11, 2012 г. – С. 68-71.
2. Чуфаров И. Особенности учета бракованной продукции в 1С. //«Системный администратор», №12, 2012 г. – С. 72-76.
3. Чуфаров И. Как избежать ошибок учета в 1С. Часть 1. Выбытие запасов. //«Системный администратор», №1-2, 2013 г. – С. 72-76.
4. Чуфаров И. Как избежать ошибок учета в 1С. Часть 2. Учет прямых материальных затрат. //«Системный администратор», №3, 2013 г. – С. 62-66.
5. Чуфаров И. Как избежать ошибок учета в 1С. Часть 3. Распределение косвенных материальных затрат. //«Системный администратор», №5, 2013 г. – С. 46-53.

6. Чуфаров И. Как избежать ошибок учета в 1С. Часть 4. Нематериальные затраты: услуги. //«Системный администратор», №6, 2013 г. – С. 52-59.
7. Чуфаров И. Как избежать ошибок учета в 1С. Часть 5. Закрытие затрат. //«Системный администратор», №7-8, 2013 г. – С. 46-52.
8. Чуфаров И. Как избежать ошибок учета в 1С. Часть 6. Спец-одежда, спецоснастка и инвентарь. //«Системный администратор», №10, 2013 г. – С. 46-53.
9. Чуфаров И. Как избежать ошибок учета в 1С. Часть 7. Учет оборудования. //«Системный администратор», №11, 2013 г. – С. 73-79.
10. Чуфаров И. Как избежать ошибок учета в 1С. Часть 8. Объекты строительства. //«Системный администратор», №12, 2013 г. – С. 38-43.
11. Чуфаров И. Как избежать ошибок учета в 1С. Часть 9. Амортизация основных средств. //«Системный администратор», №1-2, 2014 г. – С. 76-82.
12. Чуфаров И. ПАУЗ в 1С – друг или враг? Часть 1. Развенчание мифов и поиск правды. //«Системный администратор», №3, 2014 г. – С. 39-45.
13. Чуфаров И. ПАУЗ в 1С – друг или враг? Часть 2. Детальный учет несмотря ни на что. //«Системный администратор», №4, 2014 г. – С. 39-45.
14. Чуфаров И. Как избежать ошибок учета в 1С. Часть 10. События с основными средствами. //«Системный администратор», №5, 2014 г. – С. 40-45.
15. Абрашина Е.В., Емельянов И.М. Использование механизма расширенной аналитики в «1С:Управление производственным предприятием». – М.: Издательство «1С-Паблишинг», 2011. – 177 с.
16. 1С:Предприятие 8. Конфигурация «Управление производственным предприятием». Редакция 1.3. Часть 2. Учет и финансы. – М.: «Фирма 1с», 2009. – 661 с.
17. 1С:Предприятие 8. Конфигурация «Управление производственным предприятием». Редакция 1.3. Часть 3. Торговля. – М.: «Фирма 1с», 2009. – 712 с.
18. 1С:Предприятие 8. Конфигурация «Управление производственным предприятием». Редакция 1.3. Часть 4. Производство. – М.: «Фирма 1с», 2009. – 160 с.
19. Серебряков С.Л. Использование программы 1С: УПП для учета на производстве. – М.: Издательство «АРТ саТерра», 2011. – 256 с.

Ключевые слова: 1С, ПАУЗ, учет, НДС.

Рисунок 8. Проведение документов по регистрам НДС является обязательной регламентной процедурой при включенном режиме ПАУЗ





Визитка

ОЛЕГ ФИЛИПОВ, АИТ-Информ,
заместитель начальника отдела разработки, comol@mail.ru

1С:ERP 2.0. Что нового?

Часть 3. Подсистемы «Финансы» и «Бюджетирование»

В конце 2013 года на рынок вышло новое флагманское решение от 1С — «ERP 2.0», которое пришло на замену «1С:УПП». В предыдущей статье [1] были рассмотрены подсистемы «Закупки» и «Склад»

Подсистема «Финансы»

На первый взгляд подсистема финансового учета в ERP 2 была переработана относительно УПП 1.3. Начинаются изменения, наверное, с того, что подсистемы «Финансы» в УПП не было, интерфейс назывался «Управление денежными средствами», при этом он включал в себя достаточно ограниченный набор функций, которые в ERP 2 включены в подсистему финансов. На мой взгляд, данное выделение функционала в УПП 1.3 было не совсем правильно. По крайней мере мне редко приходилось встречать пользователей, которые работали бы только в интерфейсе «Управление денежными средствами». Все интерфейсы, которые доступны пользователю для переключения из УПП 1.3 совместно с подсистемами из ERP 2, приведены на рис. 1 для сравнения.

Сравнивать подсистемы с интерфейсами, возможно, не совсем корректно, но, с точки зрения пользователя, назначение данных элементов примерно схожее. Конечно, финансовый блок должен «пронизывать» всю ERP-систему

и присутствовать в том или ином виде практически в каждом учетном контуре.

В ERP 2, на мой взгляд, состав подсистемы финансов более правильный (см. рис. 2). В ней присутствуют как платежные (расчетные) операции, так и документы возникновения прибыли/убытка или дебиторской/кредиторской задолженности. Подобный подход приближает логику 1С к большинству западных ERP-систем, на которые мы привыкли ориентироваться как на мировой Best practice при решении определенных задач. К примеру, в SAP подсистема финансов (FI) включает в себя любые операции, которые отражаются на счетах бухгалтерского учета. Традиционно в популярных ERP-системах нет жесткого разделения на бухгалтерский и управленческий учет, как это сложилось в практике ведения российского бухгалтерского учета. Соответственно финансовый и бухгалтерский блок в них одно и то же. По этому пути теперь пошла и 1С. С развитием стандартов РСБУ и переходом части компаний на сдачу отчетности по МСФО данный процесс стал возможен и даже необходим.

Несмотря на это, изменений в отдельных функциях не так много, как кажется на первый взгляд. В подсистеме «Финансы» можно обнаружить функционал «закрывание месяца», который мы привыкли считать «бухгалтерским».

Из совсем нового, что появилось в данной подсистеме, стоит отметить отдельный механизм начисления процентов по кредитам и займам. Функционально он очень полезен: попробуйте хотя бы один раз вручную посчитать сложный процент с периодическими платежами и оцените всю прелесть нововведений. Хотя, конечно, за годы работы на системах 1С все уже привыкли, что кредитный договор — это договор с типом «прочее», расчеты по нему проводятся в Excel или запрашиваются у банка, а в 1С они вносятся ручными операциями. Теперь это не так, что не может не радовать.

Переработан механизм заявок на расход денежных средств с использованием новых возможностей управляемых форм. В заявке и даже в списке заявок наконец-то (!) в штатной 1С можно увидеть, оплачена она или нет. Также немаловажный функционал — это возможность согласовывать (или отклонять) заявки прямо из формы списка, а так-

Рисунок 1. Подсистемы ERP 2 и переключаемые интерфейсы УПП 1.3

Модули и планирование	Модули и планирование
- Продажи - Закупки - Склад - Производство - Финансы - Бюджетирование - Зарплата - Кадры - Внеоборотные активы - Регламентированный учет - Международный финансовый учет - Мониторинг целевых показателей - Нормативно-справочная информация - Организатор - Администрирование	- Продажи - Закупки - Склад - Производство - Финансы - Бюджетирование - Зарплата - Кадры - Внеоборотные активы - Регламентированный учет - Международный финансовый учет - Мониторинг целевых показателей - Нормативно-справочная информация - Организатор - Администрирование
- Управление персоналом - Кадровый учет организаций - Расчеты с персоналом - Расчет зарплаты организаций - Бухгалтерский и налоговый учет - Международный учет - Заведующий учетом - Администрирование пользователей	

же отдельный список неподтвержденных заявок. Теперь действительно можно пользоваться штатным механизмом планирования денежных средств. По крайней мере в достаточно простых ситуациях. Кроме того, не стоит забывать, что заявки на расходование денежных средств – это инструментарий, который должен быть практически у всех сотрудников компании, даже у тех, кто не работает непосредственно в 1С:ERP. Руководствуясь этими же соображениями, можно еще большим плюсом в данном вопросе назвать тонкий и особенно веб-клиент.

С документами по кассе и банку существенных изменений не произошло, только переделали на управляемые формы. Да и вряд ли в этом блоке могли потребоваться существенные переделки.

Закрытие месяца в ERP 2 больше похоже на сходную операцию в бухгалтерии, чем на сложный и запутанный инструмент УПП 1.3. Мое личное мнение, что упрощения, которые были предприняты в ERP 2, пойдут на пользу программному продукту такого класса. Внешний вид закрытия месяца в ERP 2 представлен на рис. 3.

Подсистема «Бюджетирование»

Подсистему «Бюджетирование» в ERP 2 можно в целом назвать «полностью новой». С тем, что было в УПП 1.3, данный

функционал имеет мало общего. Я лично сильно этому порадовался. Те читатели, кто хоть раз пытался или даже внедрил бюджетирование на базе УПП 1.3, меня поймут. Теперь можно сказать, что ERP по функционалу контура бюджетирования практически вплотную приблизилась к «лидерам» данного рынка среди систем на платформе 1С в России – к «1С:Интелев» и «Бит:Финанс». А если учесть полностью открытый код, отсутствие дополнительных компонентов защиты и традиционно более качественную технологическую проработку решения (относительно решений партнеров), то уже сейчас многие, наверное, предпочли бы ERP 2 в качестве системы бюджетирования. Состав подсистемы бюджетирования не слишком обширный (см. рис. 4).

Появилась сущность «модель» – это, по сути своей, все настройки подсистемы бюджетирования, которые вы можете сделать. Сквозная аналитика. Модель нужна для того, чтобы объединять усилия разных консультантов одной системы или, наоборот, разъединять... Но сама форма модели не особенно интересна, зато есть гиперссылка, из которой попадаем в другой механизм «шаги бюджетного процесса», которым уже задаются некоторые маршруты. Данная настройка процесса очень важна.

Форма шагов бюджетных процессов представлена на рис. 5.

Рисунок 2. Состав подсистемы «Финансы» ERP 2



Рисунок 3. Внешний вид закрытия месяца в ERP 2

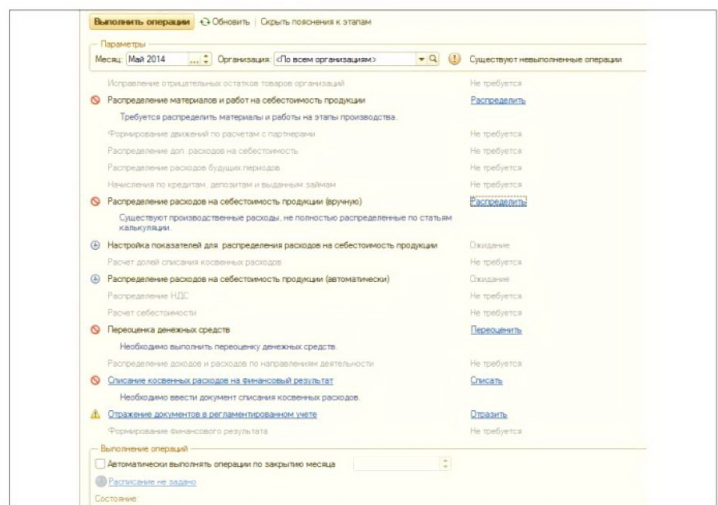


Рисунок 4. Состав подсистемы бюджетирования

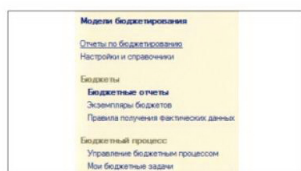
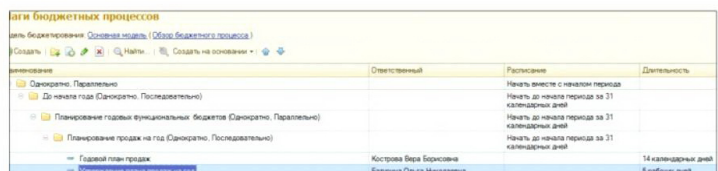


Рисунок 5. Форма настройки шагов бюджетного процесса



Появилась в ERP 2 наконец такая сущность, как «Вид бюджета», сущность, которая есть, по-моему, во всех системах бюджетирования наравне со «статьями бюджета» – не видели мы ее, по-моему, только в ERP. Но «Вид бюджета» – не просто аналитический разрез. В форму данного справочника как раз заносятся правила получения данных, их пересчета и нормализации. Внешне инструмент произ-

водит весьма положительное впечатление. Но вопрос: получится ли обучить подобным настройкам рядового пользователя? Особенно «Вид бюджета» интересно смотреть при открытии формы элемента. На рис. 7 представлен внешний вид формы «Виды бюджетов».

Собственно, сущность «Вид бюджета» в том числе используется и в отчетах. Практически тот же самый функционал для более широкого круга задач, чем просто трансляция факта. По сути, в настройках для вида бюджета задаются алгоритм его формирования, алгоритм трансляции факта и алгоритм план-фактного анализа.

Сущность «Экземпляры бюджетов» (см. рис. 8) – документ, которым вводятся привычные плановые данные. Документ достаточно интересный и заслуживает отдельного внимания. До этого решения (речь о ERP 2) 1С старательно избегала использования табличного документа в режиме ввода данных. Подобного мы не видели даже в решениях «1С:Совместимо», не говоря уже о типовых. Впрочем, в «1С:Совместимо» скорее всего просто не прошли бы продукты с такой функциональностью. Табличный документ в режиме ввода данных использовался в решениях на 1С 7.7. Но там просто не было альтернатив ввиду скудности функционала табличного поля. Теперь, похоже, история пошла по спирали. Традиционно тяжелый функционал ввода данных бюджетирования решено было сделать именно так.

Рисунок 6. Управление бюджетным процессом

Рисунок 7. Внешний вид формы «Виды бюджетов»

Рисунок 8. Документ «Экземпляры бюджетов»

Статья бюджетов / Аналитика	2013 г.	Март 2013 г.	Апрель 2013 г.	Май 2013 г.	Июнь 2013 г.	Июль 2013 г.	7
Закупки товара	2	3	4	5	6	7	
Ключ шестигранный 5x4	2	3	4	5	6	7	
набор ключей комбинированных с трехсторонней Матри 8-19 мм 7 шт							
набор ключей наконечных Матри 6-32 мм 8 шт полированный хром							
набор ключей рожковых Матри 6 x 17 мм 6 шт полированный, хромированный							
набор отверток Stanley Basic из 19 шт							
набор отверток Stanley Fatmax TORX из 6 штук							
«прочие номенклатура»							
Возврат товара поставщикам (Итого (сумма))	2	3	4	5	6	7	

Возможно, в этом есть свои плюсы. Данный документ (вернее, его табличная часть) может полностью изменить свой внешний вид в зависимости от вида бюджета, в нем выбранного. Выбирать, конечно, можно только из тех бюджетов, для которых установлен флаг «Используется для ввода плана», то есть для первичных данных. Но самое интересное, что внешний вид документа и правила пересчета одних ячеек от других полностью настраиваемы. В частности, для них действуют все те же правила настройки, что и для любого другого вида бюджета (см. рис. 7), но часть данных при этом можно ввести вручную. По сравнению с УПП, где все эти данные вводились документами «Бюджетная операция», это не просто шаг, а целый большой прыжок вперед.

Подсистема «Бюджетирование» в ERP просто изобилует «красивыми» элементами. Это несколько нестандартно для типовых конфигураций 1С. Как правило, реализация подобных функций ложилась на плечи партнеров.

Относительно УПП появился красивый и удобный инструмент «Управление бюджетным процессом». Он представляет собой диаграмму Ганта с задачами, которые относятся к бюджетированию. Вести эти задачи в системе, в которой они непосредственно исполняются, наверное, правильно и логично. В любом случае, посмотреть (см. рис. 6) на этот инструмент, наверное, интересно.

И, наверное, самое важное и самое спорное (на мой взгляд) новшество ERP 2 относительно УПП. В ERP 2 нет плана счетов бюджетирования и нет регистра бухгалтерии для бюджетирования. Все бюджетные данные должны собираться из оперативного контура. Ответ на вопрос, поче-

му, простой – разработчики не любят регистр бухгалтерии. Слишком много проблем вызывает необходимость записи в него оперативных данных, они решили, что для бюджетирования он будет лишним. Но, на мой взгляд, это не совсем верно. Часто в контуре бюджетирования компании ведут также управленческий учет, потому как он может существенно не совпадать с бухгалтерским в российских компаниях. Кроме того, это значительно упрощает план-фактный анализ. Оперативно эти данные не особо нужны. Они скорее аналитические, поэтому механизма трансляции было бы достаточно. А сам инструмент проводок нужен для того, чтобы удобно и корректно отразить двойную запись и в итоге собрать баланс. Теперь же эта процедура существенно усложнится.

Итого, подсистема финансов доработана не слишком сильно, но при этом все изменения нужные и очень полезные. Подсистема «Бюджетирование» написана заново. Функционал очень мощный, интересный, удобный. Есть только один главный и существенный минус – отсутствие плана счетов в контуре бюджетирования. В следующей статье будут рассмотрены подсистемы «Зарплата», «Кадры», «Внеоборотные активы», «Регламентированный учет». **БОБ**

1. Филиппов О. 1С:ERP 2.0. Что нового? Часть 2. Подсистемы «Закупки» и «Склад». // «Системный администратор», №5, 2014 г. – С. 37-39 (<http://samag.ru/archive/article/2689>).

Ключевые слова: 1С:ERP, учет, финансы, бюджетирование.

Выбирайте прямой путь! Подписывайтесь в редакции!

Что вы выигрываете?

1. Вы экономите деньги – редакционная подписка самая дешевая из возможных вариантов
2. Вы экономите время – подписка оформляется в режиме он-лайн
3. Журнал мы отправляем заказными письмами, что сводит риск неполучения до минимума. В случае потери письма почтой мы направим вам номер повторно

Стоимость редакционной подписки:

Смешанная версия (печатная + электронная)

На год – 5040 р, на полгода – 2520 р

Печатная версия

На год – 4080 р, на полгода – 2040 р

Электронная версия

На год – 1500 р, на полгода – 750 р

E-mail: sa@samag.ru Tel.: (499) 277-12-41 Fax: (499) 277-12-45





Визитка

СЕРГЕЙ ИЛЬИЧЕВ, ЧОУ «ТПКГ», г. Тула, системный администратор и учитель информатики, sergil68@mail.ru

Программирование USB в Android

Используем для связи интерфейс USB

Попробуем разобраться в том, как организовать обмен информацией между различными (в том числе и самостоятельно разработанными) электронными устройствами и аппаратами (планшетами или телефонами), работающими под управлением операционной системы (ОС) Android с использованием USB

Для того чтобы не запутать читателя, в статье первые будут называться просто «устройства», а вторые – «мобильные устройства».

Данная статья адресована прежде всего тем, у кого есть какое-либо электронное устройство (разработанное самостоятельно или кем-то еще), протокол обмена данными с которым хорошо известен (например, уже есть программа, работающая с этим устройством в ОС Windows/Linux), и хотелось бы иметь программу, работающую с ним еще и в Android.

USB и Android

Уже сравнительно давно, начиная с Android Honeycomb (версия 3.1), в ОС от Google появилась возможность использовать мобильное устройство в режиме хоста (Host mode), в котором оно питает шину и может определять подключенные устройства с помощью прикладного программного интерфейса (API) на языке Java (см. [1] и рис. 1).

Устройство, работающее в хост-режиме, является инициатором передачи данных и управляет процессом обмена информацией по каналу связи.

USB Host API появился в Android в середине 2011 года, однако информации по его использованию на русском языке в сети до сих пор очень мало. Интерес к этому вопросу подогревает наличие на рынке мобильных устройств, на которых работает этот API, стоимостью менее 3000 руб. (см., например, тестирование в [2]).

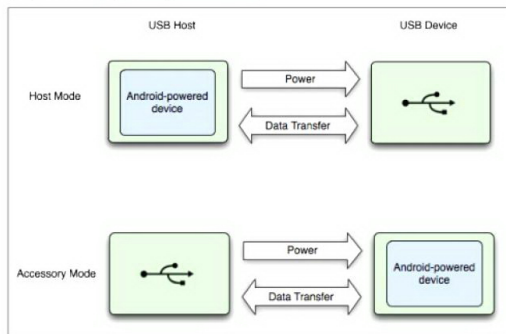
Отметим, что USB – не единственный способ связи с тем же самодельным устройством. Android позволяет использовать еще Bluetooth, NFC, Wi-Fi P2P, SIP, а также стандартное сетевое подключение [3]. Так что в арсенале разработчика достаточно возможностей для осуществления своих самых смелых замыслов.

Другим распространенным вариантом связи с различными устройствами до сих пор является использование переходника USB-COM. Материал в сети по применению переходника USB-COM в Android есть – см., например, [4]. Популярность такого подключения обусловлена наличием большого количества уже разработанных с использованием различных микроконтроллеров устройств, связь с которыми осуществляется с помощью COM-порта (последовательного порта), что 10 лет назад являлось почти стандартным способом передать данные от компьютера к самодельной железяке.

В сравнении с COM-портом использование USB позволяет существенно повысить скорость передачи данных и сделать этот процесс удобным для пользователя. Скорость передачи, которая даже в случае низкоскоростных устройств (клавиатуры, мыши, джойстики) составляет 10-1500 Кбит/с, простота и невысокая стоимость кабельной системы и подключений, самоидентификация устройств с автоматическим конфигурированием, скрытие подробностей электрического подключения от конечного пользователя (плюс возможность отключения кабеля без выключения устройств), контроль ошибок и их восстановление на уровне протокола – вот неоспоримые преимущества данной технологии (см. [4], с. 12).

Вообще, говоря об использовании USB для передачи данных, не лишним будет упомянуть книгу П.Агурова «Интерфейс USB» [5].

Рисунок 1. Иллюстрация работы Android-устройства в режимах USB Host и Accessory (рисунок с сайта <http://developer.android.com>)



Она, хотя часто и критикуется в сети и выпущена последний раз в 2006 году, не раз помогла найти верное решение при поиске информации по различным аспектам применения этой технологии. В книге рассмотрены вопросы от выбора микросхемы и схемотехники для контроллера до написания программы микроконтроллера и примеров программирования передачи данных по протоколу USB со стороны компьютера.

Нельзя не указать и первоисточник данных по этому вопросу – сайт некоммерческой организации USB IF (USB Implementers Forum), занимающейся разработкой спецификации этого интерфейса [6], правда, данный материал на английском языке. Однако именно там вы найдете исчерпывающие сведения об устройстве интерфейса USB. Есть неплохой перевод частей спецификации [7]. Интересующимся программными решениями со стороны микроконтроллера также можно посмотреть ссылку [8].

Немного о классах USB-устройств

Необходимо отметить, что разработка программного обеспечения для обмена данными с конкретным устройством сильно зависит от его реализации на уровне микроконтроллера.

Привести примеры программ связи для всех типов USB-устройств в рамках одной статьи по понятным причинам невозможно (начальные сведения о программировании различных типов устройств можно почерпнуть в [5]). Однако мы ограничимся тем, что приведем код, реализующий поиск нужного устройства и доступ к его контрольным точкам для обмена информацией.

Также разберем отправку данных на примере одного из типов USB-устройств, а именно класса устройств HID (human interface device – класс устройств для взаимодействия с человеком). Этот класс включает в себя «медленные» устройства, такие как клавиатура, мышь, джойстик, и примеров его реализации с помощью различных микроконтроллеров в сети достаточно (есть, например, и в [8]).

Почему именно класс HID так полюбился изготовителям различных самодельных устройств? Прочитав Википедию [9]: «Помимо детальных спецификаций классических устройств ввода (типа клавиатур и мышек), стандарт HID определяет особый класс устройств без детальных спецификаций. Этот класс именуется USB HID Consumer Control и представляет собой, по сути, нерегламентированный канал связи с устройством.

При этом устройство пользуется теми же стандартными для операционной системы драйверами, что и мышка с клавиатурой. Таким образом, можно создать USB-устройство, которое не требует создания и установки специальных драйверов в большинстве распространенных компьютерных операционных систем». Остается добавить только, что работает эта спецификация и в ОС Android (не исключая прошивку CyanogenMod).

Одним из вариантов обмена данными с HID-устройством является передача по прерываниям (interrupt transfer), которая используется в том случае, когда необходимо передать пакеты данных небольшого размера (максимальный размер пакета зависит от скорости передачи и составляет от 64 до 1024 байт) через заданный временной интервал. Пакет для передачи называется репортом (англ. – report, см. [5], с. 71, 95).

Такой длины репорта обычно вполне хватает для обмена информацией с самодельным устройством. Например, 64 байта информации в одном пакете – это довольно много для контроллера, ведь для передачи состояний светодиода или простейшего датчика достаточно 1 бита информации.

Необходимые инструменты

Итак, нам понадобятся планшет или телефон с Android-версией не ниже 3.1. Здесь необходимо отметить, что вышеуказанный USB Host API полностью реализован не на всех мобильных устройствах (об этом упоминается и на сайте developer.android.com [1]).

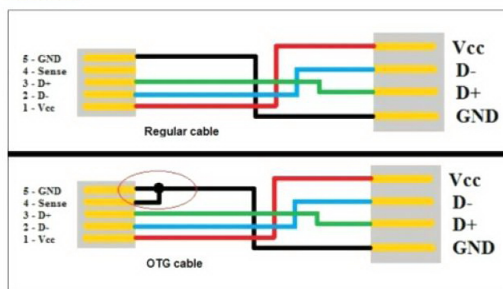
В некоторых планшетах/телефонах разъем USB используется только для зарядки и связи с персональным компьютером. Еще раз отправлю читателя к списку мобильных устройств, пригодных или непригодных для наших опытов [2].

Понадобятся также какое-либо USB-устройство (для первых опытов будет достаточно обычного USB-флеш-накопителя), переходник OTG (On-The-Go – см. рис. 2) и/или шнур USB для связи с устройством. В Википедии по поводу OTG говорится: «При подключении через USB OTG ранг устройства (ведущий или ведомый) определяется наличием или отсутствием перемычки между контактами 4 и 5 в штекере соединительного кабеля. В USB OTG-кабеле такая перемычка устанавливается лишь в одном из двух разъемов [10]». Соответственно нам необходима такая перемычка со стороны мобильного устройства.

Неплохим подспорьем в работе будет также программа USB Device Info, установленная из хранилища Google Play Market. Программа умеет определять подключенные к USB-разъему планшета/телефона устройства как с помощью Java API, так и с помощью ядра Linux. То есть если ваше устройство не определилось с помощью Java USB Host API в USB Device Info, то с большой вероятностью точно будет использоваться для этого мобильного устройства какую-либо (в том числе и свою) Android-программу, написанную с помощью Java и USB Host API.

Иногда очень полезной бывает информация, выводимая командой lsusb операционной системы Linux. С ключами -v и -d lsusb выводит о USB-устройстве все, или почти все, что необходимо разработчику программного обеспечения для устройств этого класса.

Рисунок 2. Различия в схеме обычного USB-кабеля и OTG-кабеля (рисунок с сайта tech.firstpost.com)



Далее необходим компьютер с установленным Android SDK и интегрированной средой разработки (IDE) Eclipse с плагином ADT (хотя можно обойтись и только SDK). Как создать и установить приложение для Android, можно посмотреть, например, в [11, 12] или в сети Интернет. Ну и, конечно, необходимо хотя бы минимальное знание языка программирования Java, а также желание добиться результата, без него никак!

Отмечу, что на выяснение некоторых технических вопросов применения USB в Android автору потребовались недели кропотливого поиска информации.

Классы Java для работы с USB в Android API

Итак, как говорится на сайте разработчиков USB Host API для Android [13], «прежде чем начать, важно понять, какие классы вы будете использовать в работе». В таблице 1 приведено описание важнейших классов для работы с USB Host API.

Практически во всех случаях применения USB Host API программист использует эти классы в своей работе. Алгоритм их применения выглядит примерно так: определяем устройства (цель – программный доступ к классу `UsbDevice`), подключенные к хосту (мобильному устройству), с помощью `UsbManager`. Когда программный доступ к устройству получен, необходимо определить соответствующие `UsbInterface` и `UsbEndpoint` для общения с ним. Как только вы получили в свое распоряжение конечную точку, откройте `UsbDeviceConnection`, чтобы общаться с USB-устройством.

Если конечная точка работает в режиме асинхронной передачи, используем класс `UsbRequest`.

Таблица 1. Описание классов для работы с USB в Android

Название класса	Описание
<code>UsbManager</code>	Позволяет определять подключенное USB-устройство и обмениваться с ним данными
<code>UsbDevice</code>	Представляет подключенное USB-устройство и содержит методы для доступа к его идентификационной информации, интерфейсам, и конечным точкам
<code>UsbInterface</code>	Представляет «интерфейс» USB-устройства, который определяет набор функций для данного устройства. Одно устройство может иметь один или несколько интерфейсов для обмена информацией
<code>UsbEndpoint</code>	Представляет «конечную точку» интерфейса, которая и является каналом связи для этого интерфейса. Интерфейс может иметь одну и более конечных точек и обычно имеет конечные точки для получения информации и для ее передачи
<code>UsbDeviceConnection</code>	Представляет «подключение» к данному устройству. Необходимо для передачи данных в конечную точку. Этот класс позволяет получать данные или передавать синхронно или асинхронно
<code>UsbRequest</code>	Представляет асинхронный запрос на обмен данными с устройством через <code>UsbDeviceConnection</code>
<code>UsbConstants</code>	Определяет константы, которые соответствуют определениям в <code>linux/usb/ch9.h</code> ядра Linux

Давайте попробуем со всем этим разобраться, создав простое приложение, которое, используя этот API, определит подключенное к хосту с ОС Android устройство и выведет о нем некоторую информацию на экран телефона или планшета.

Создаем проект

В Eclipse проект создается с помощью пунктов меню `File → New → Android Application Project`. Заметим также, что код, приведенный ниже, заимствован из приложений-примеров, поставляемых с Android SDK (папка `android sdk samples/ android-N(API Level)/USB`), речь идет о программе управления USB-игрушкой `Missile Launcher`.

Примеры приложений загружаются через Android SDK Manager (нужно отметить пункт `Samples for SDK [11]`). В листингах, приведенных ниже, примеры кода снабжены комментариями, которые объясняют суть происходящего.

Создавая проект, не забудьте отметить нужный API Level в опции `Minimum Required SDK (API Level 12, соответствующий версии Android 3.1 /Honeycomb/ или выше)`. В проекте будет очень простой интерфейс пользователя – главное окно (`Activity`) и `TextView` для вывода информации. Подобный проект подробно рассмотрен в [12].

В созданном автоматически классе для `Activity` нашего проекта необходимо определить следующие экземпляры классов для работы с USB:

```
private TextView lgView;
private UsbManager mUsbManager;
private UsbDevice mDevice;
private UsbDeviceConnection mConnection;
private UsbEndpoint mEndpointIntr;
```

Далее в обработчике `onCreate()` `Activity` создадим `TextView` для записи сообщений нашей программы (также не забываем указать об этом в `xml`-файле разметки в `res/layout`):

```
lgView = (TextView)findViewById(R.id.logTextView);
```

и получаем доступ к классу `UsbManager`:

```
mUsbManager = (UsbManager) getSystemService(
    Context.USB_SERVICE);
```

Создадим еще обработчик события `onResume()`. Наша цель – чтобы информация о подключенных устройствах обновлялась при активизации окна нашего приложения (см. листинг 1).

Листинг 1. Обработчик события `onResume()`

```
public void onResume() {
    super.onResume();
    //заполняем контейнер списком устройств
    HashMap<String, UsbDevice> deviceList =
        mUsbManager.getDeviceList();
    Iterator<UsbDevice> deviceIterator =
        deviceList.values().iterator();

    lgView.setText("Devices Count:" + deviceList.size());

    while (deviceIterator.hasNext()) {
        UsbDevice device = (UsbDevice) deviceIterator.next();
        //пример определения ProductID устройства
        lgView.setText(lgView.getText() + "\n" +
            device.getProductID());
    }
}
```

```

        "Device ProductID: " + device.getProductID() );
    }

    //определяем намерение, описанное в фильтре
    // намерений AndroidManifest.xml
    Intent intent = getIntent();
    lgView.setText( lgView.getText() + "\n" + "\n"
        "intent: " + intent);
    String action = intent.getAction();

    //если устройство подключено, передаем ссылку
    //в функцию setDevice()
    UsbDevice device =
        (UsbDevice)intent.getParcelableExtra( "
        UsbManager.EXTRA_DEVICE);

    if (UsbManager.ACTION_UsbDeviceAttached.equals(action)) {
        setDevice(device);
        lgView.setText( lgView.getText() + "\n" + "\n"
            "UsbManager.ACTION_USB_DEVICE_ATTACHED.
            equals(action) is TRUE");
    } else if (UsbManager.ACTION_USB_DEVICE_DETACHED.equals(action)) {

        if (mDevice != null && mDevice.equals(device)) {
            setDevice(null);
            lgView.setText( lgView.getText() + "\n" + "\n"
                "UsbManager.ACTION_USB_DEVICE_DETACHED.equals(
                action) is TRUE");
        }
    }
}
}

```

Далее для Activity создадим функцию `setDevice()`, необходимую для работы с нашим устройством (см. листинг 2). В обработчике `onResume()` и в функции `setDevice()` мы в точности выполнили алгоритм применения USB Host API, описанный в предыдущем разделе.

Листинг 2. Функция `setDevice()`

```
private void setDevice(UsbDevice device) {
    lgView.setText( lgView.getText() + "\n"
        + "\n" + "setDevice " + device);
    //определяем доступные интерфейсы устройства
    if (device.getInterfaceCount() != 1) {
        lgView.setText( lgView.getText() + "\n"
            + "\n" + "could not find interface");
        return;
    }

    UsbInterface intf = device.getInterface(0);
    //определяем конечные точки устройства
    if (intf.getEndpointCount() == 0) {
        lgView.setText( lgView.getText() + "\n"
            + "\n" + "could not find endpoint");
        return;
    } else {
        lgView.setText( lgView.getText() + "\n" + "\n"
            + "Endpoints Count: " + intf.getEndpointCount() );
    }

    UsbEndpoint epIN = null;
    UsbEndpoint epOUT = null;

    //ищем конечные точки для передачи по прерываниям
    for (int i = 0; i < intf.getEndpointCount(); i++) {
        if (intf.getEndpoint(i).getType() == UsbConstants.USB_ENDPOINT_XFER_INT) {
            if (intf.getEndpoint(i).getDirection() == UsbConstants.USB_DIR_IN) {
                epIN = intf.getEndpoint(i);
            } else {
                epOUT = intf.getEndpoint(i);
            }
        }
    }
}
```

```
else {  
    epOUT = intf.getEndpoint(i);  
    lgView.setText( lgView.getText() + "\n" + "  
        "OUT endpoint: " + intf.getEndpoint(i) );  
}  
  
} else {  
    lgView.setText( lgView.getText() + "\n" + "  
        "no endpoints for INTERRUPT_TRANSFER"); }  
}  
  
mDevice = device;  
mEndPointIntr = epOUT;  
//открываем устройство для передачи данных  
if (device != null) {  
    UsbDeviceConnection connection = "  
        mUsbManager.openDevice(device);  
  
    if (connection != "  
        null && connection.claimInterface(intf, true)) {  
            lgView.setText( lgView.getText() + "\n" + "  
                "open device SUCCESS!");  
            mConnection = connection;  
  
        } else {  
            lgView.setText( lgView.getText() + "\n" + "  
                "open device FAIL!");  
            mConnection = null;  
        }  
    }  
}
```

В дополнение к приведенному коду, который, как уже наверняка догадался внимательный читатель, открывает устройство для приема-передачи данных, остается лишь задействовать протокол обмена данными, который, повторюсь, должен быть хорошо известен разработчику.

Приведем лишь, как было обещано, код, который отправит HID-устройству некоторый пакет данных `message`, используя передачу по прерываниям, класс `UsbRequest`, и соответствующую конечную точку (см. листинг 3).

Листинг 3. Пример кода для отправки данных устройству

```
//определение размера буфера для отправки
//исходя из максимального размера пакета
int bufferDataLength = mEndpointIntr.getMaxPacketSize();
lgView.setText( lgView.getText() + "\n" + mEndpointIntr.getMaxPacketSize() );

ByteBuffer buffer = \
    ByteBuffer.allocate(bufferDataLength + 1);
UsbRequest request = new UsbRequest();

buffer.put(message);
request.initialize(mConnection, mEndpointIntr);
request.queue(buffer, bufferDataLength);
try
{
    if (request.equals(mConnection.requestWait()))
    {
        //отправка прошла успешно
        //lgView.setText( lgView.getText() + "\n" +
        //    "sending CLEAR!!!");
    }
}

catch (Exception ex)
{
    //что-то не так...
    //lgView.setText( lgView.getText() + "\n" +
    //    "sending not clear...");
}
```


Фильтрация устройств в AndroidManifest.xml

Хотя в нашем приложении нет нужды в поиске конкретного устройства с известными VID (Vendor-ID) и PID (Product-ID), инженеры Google не приводят примеров приложений без секции `<intent-filter>` в манифест-файле, и автору не удалось добиться работы программы без фильтрации устройств в `AndroidManifest.xml` [13].

Напомним, что Vendor-ID и Product-ID – это уникальные идентификаторы USB-устройств. То есть, используя фильтрацию, можно создать приложение, которое взаимодействует лишь с определенным устройством или каким-то классом устройств. Отметим, что производители устройств должны согласовать эти номера с организацией USB IF.

Приложение, манифест-файл которого приведен в листинге 4, а файл с условием фильтрации – в листинге 5, например, с успехом распознает подключенные к мобильному устройству USB-флеш-накопители, но не распознает клавиатуру и мыши, имеющиеся у автора. Это приложение вместе с исходным кодом можно скачать по ссылке [14].

Листинг 4. Файл `AndroidManifest.xml`

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="ru.learn2prog.usbhostexample"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="12"
        android:targetSdkVersion="14" />

    <uses-feature android:name="android.hardware.usb.host" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".ru.learn2prog.usbhostexample.MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.DEFAULT" />
            </intent-filter>

            <intent-filter>
                <action android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED" />
            </intent-filter>

            <meta-data android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED"
                android:resource="@xml/device_filter" />
        </activity>
    </application>
</manifest>
```

Листинг 5. Файл фильтра `device_filter.xml` (каталог `/res/xml`)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <usb-device />
</resources>
```

Операции по сборке и установке нашего приложения ничем не отличаются от обычных. Хочу обратить внимание на действия фильтра намерений – при подключении устройства к хосту ОС запрашивает пользователя о запуске нашего приложения.

Итак, в статье мы рассмотрели вопросы, связанные с поиском устройств, подключенных по интерфейсу USB к мобильному устройству, а также пример кода для обмена данными по протоколу USB между хостом и другим устройством, используя передачу по прерыванию и класс HID. Прошу искушенного читателя простить возможные неточности и ошибки в применении технических терминов спецификации USB, так как теоретические знания в наше время не всегда успевают за практической потребностью, а, как уже отмечалось, информации по этому вопросу на русском языке не так много. Напомним, статья – накопленный практический опыт автора, которым хотелось поделиться. Было бы интересно узнать, как используют интерфейс USB для передачи данных другие программисты. **EOF**

1. USB Host and Accessory – <http://developer.android.com/intl/ru/guide/topics/connectivity/usb/index.html>.
2. Список мобильных устройств, протестированных для работы с USB Host API – www.learn2prog.ru/paracels-m.
3. Android API Guides – <http://developer.android.com/intl/ru/guide/topics/connectivity/index.html>.
4. Использование переходника USB-COM для связи с устройством в Android – <http://habrahabr.ru/post/163913>.
5. Агуров П. Интерфейс USB. – 2005 г., BHV-СПб ISBN: 5-94157-202-6.
6. Сайт организации USB IF – <http://www.usb.org>.
7. Чекунов Д. Программисту USB-устройств. Части 1, 2, 3 – <http://www.soel.ru/issues/?id=299302>.
8. USB Framework for PIC18, PIC24 & PIC32 – http://www.microchip.com/stellent/idcplg?dclid=SS_GET_PAGE&nodelid=2680&dDocName=en537044.
9. Википедия – USB HID – http://ru.wikipedia.org/wiki/USB_HID.
10. Википедия – USB OTG – http://ru.wikipedia.org/wiki/OTG#USB_OTG.
11. Ильичев С. Программирование для Android? Это просто! Часть 2. // «Системный администратор», №7-8, 2013 г. – С. 78-81.
12. Ильичев С. Программирование для Android. Изучаем ресурсы и делаем приложение в ADT. // «Системный администратор», №11, 2013 г. – С. 46-50.
13. Обзор классов, необходимых для работы с USB в Android – <http://developer.android.com/guide/topics/connectivity/usb/host.html>.
14. Ссылка на исходники приложения – <http://www.learn2prog.ru/android-prog-usb>.

Ключевые слова: Android, программирование, USB.



**5-ая Международная
конференция
WEB-разработчиков**

Москва, 14 июня 2014

Традиционно приезжают авторы языков
разработки и гуру!

ПРИГЛАШАЕМ ДОКЛАДЧИКОВ!

Основные направления разработки

PHP, Python, Ruby, @Mobi, JavaScript, Java, MySQL,
PostgreSQL, Sphinx, Tarantul, GO, Haskell, Erlang

До встречи на DEVCONF 2014!

www.devconf.ru



Визитка

АЛЕКСАНДР КАЛЕНДАРЕВ, РБК Медиа,
программист, akalend@mail.ru

Очереди. Теория и практика

В майском номере [1] было рассмотрено взаимодействие разных потоков с использованием очередей в рамках одного адресного пространства. А как организовать взаимодействие для нескольких процессов, даже если они разнесены на разные физические серверы?

Без сетевого взаимодействия тут никак не обойтись. И что в данном случае является связующим звеном между взаимодействующими компонентами? Это либо специализированная сетевая библиотека, либо выделенный процесс, именуемый сервером очередей. В данной статье будут рассмотрены различные способы использования очередей.

Очередь – это структура данных, которая соответствует принципу: первый пришел, первый вышел.

Вот несколько архитектурно-важных причин использования очередей:

Масштабируемость – очереди сообщений позволяют распределить порции информации между несколькими системами обработки. Если система не справляется с входящим потоком информации, мы можем, не изменяя архитектуру, нарастить элементы обработки информации.

Отказоустойчивость – очереди сообщений позволяют отделить обрабатывающие информацию процессы таким образом, что если некий процесс выйдет из строя, то сообщения могут быть добавлены в очередь и могут быть обработаны позднее, после восстановления.

Снижение пиковой нагрузки – процессам посредством очереди предоставляют возможность асинхронной обработки данных, не влияя на систему в целом, если процесс не успевает обработать всю приходящую информацию, то он может ее обработать позднее, используя очередь сообщений, как буфер информации.

Давайте приведем вам несколько примеров использования очередей:

- > отложенная обработка пользовательского контента;
- > передача статистики;
- > сглаживание нагрузки;
- > выполнение периодических задач;
- > оповещение процессов о некотором событии.

Существует всего несколько подходов к организации очередей:

- > использовать реляционные базы данных;
- > применить существующие библиотеки;

> использовать существующие решения (серверы очередей, NoSQL-хранилища).

Использование реляционной СУБД

Давайте дадим ответ на вопрос: а как в качестве очереди можно использовать простую БД?

Представим блокнот, в который мы записываем будущие покупки. И как только какую-то покупку осуществили, то сразу эту строчку вычеркиваем из нашего блокнота. А теперь наш воображаемый блокнот заменим на таблицу БД. В данной таблице (queue) должны как минимум быть поля:

id (autoincrement) – номер записи;

data – пользовательские данные.

Если мы хотим добавить данные в очередь, то должны вставить данные:

```
INSERT INTO queue (data) VALUES ('bla-bla-bla');
```

Примечание: если мы используем в качестве БД MySQL, то поле id должно быть long integer autoincrement. Если БД не поддерживает тип autoincrement, как, например, PostgreSQL, то для реализации autoincrement используем последовательности. По умолчанию будем понимать, что в качестве БД используется MySQL.

Если мы хотим взять элемент из хвоста очереди, то мы должны сделать две вещи:

- > взять элемент с минимальным id;
- > уничтожить элемент с выбранным id.

Чтоб не случилось такого, что, пока мы брали элемент с минимальным id, из-за конкуренции за ресурсы другой процесс взял другой элемент и минимальный id изменился, необходимо использовать транзакции:

```
START TRANSACTION;
```

```
SELECT id,data FROM queue ORDER BY id desc LIMIT 1;  
DELETE queue WHERE id=...;
```

```
COMMIT;
```

Очень часто вместо удаления можно использовать флаг isPop = 1:

```
START TRANSACTION;

SELECT id,data FROM queue WHERE is pop=0
ORDER BY id desc LIMIT 1;
UPDATE queue SET isPop = 1 WHERE id=...;

COMMIT;
```

Недостаток использования БД такой же, как и у любой БД, – невысокая производительность.

Достоинство: простота использования.

Использование специальных библиотек. OMQ

Если осуществляется разработка программного обеспечения, в которой при обмене сообщениями требуется технология уведомлений, то можно рекомендовать использовать библиотеку OMQ (ZeroMQ) [2, 3].

OMQ – одна из самых производительных библиотек и может использовать многоадресный транспортный протокол, который является эффективным методом для передачи данных в различных направлениях.

Давайте рассмотрим простой тестовый пример, в котором у нас есть сервер и клиент, который отправляет на сервер сообщение hello, а сервер, в свою очередь, отвечает на него сообщением: world. Сервер слушает порт 4040, а клиент соответственно отправляет на этот порт сообщения.

Ниже приведен пример кода сервера, который по приходу сообщения hello, отправляет обратно клиенту сообщение world:

```
#include <string.h>
#include <stdio.h>
#include <unistd.h>
#include "zmq.h"

// объявляем контекст OMQ и OMQ сокет
int main (int argc, char const *argv[]) {
    void* context = zmq_ctx_new();
    void* respond = zmq_socket(context, ZMQ_REP);

    // OMQ слушает сообщения на порту 4040
    zmq_bind(respond, "tcp://*:4040");

    // в цикле ожидаем сообщения
    for(;;)
    {
        zmq_msg_t request;
        // инициализация сообщения
        zmq_msg_init(&request);
        // прием сообщения
        zmq_msg_rcv(&request, respond, 0);
        printf("Received: hello\n");
        // закрываем сообщение, освобождаем ресурсы
        zmq_msg_close(&request);

        // sleep one second
        sleep(1);
        zmq_msg_t reply;
        // инициализация ответа на 5 байт
        zmq_msg_init_size(&reply, strlen("world"));
        // копируем само сообщение в область ответа
        memcpy(zmq_msg_data(&reply), "world", 5);
        // отправляем ответ
        zmq_msg_send(&reply, respond, 0);
        // закрываем ответ, освобождаем ресурсы
        zmq_msg_close(&reply);
    }
}
```

```
// закрываем контекст
zmq_close(respond);
// освобождаем ресурсы
zmq_ctx_destroy(context);
return 0;
}
```

Ниже приведен код клиента, который посылает сообщение hello на сервер:

```
#include <string.h>
#include <stdio.h>
#include <unistd.h>
#include "zmq.h"

int main (int argc, char const *argv[]) {
    // объявляем контекст OMQ и OMQ сокет
    void* context = zmq_ctx_new();
    void* request = zmq_socket(context, ZMQ_REQ);
    //соединяется с OMQ сервером

    zmq_connect(request, "tcp://localhost:4040");
    int count = 0;

    for(;;)
    {
        zmq_msg_t req;
        // инициализация сообщения, выделение ресурсов
        zmq_msg_init_size(&req, strlen("hello"));
        //копирование сообщения в область передачи
        memcpy(zmq_msg_data(&req), "hello", 5);
        printf("Sending: hello - %d\n", count);
        // отправляем сообщение
        zmq_msg_send(&req, request, 0);
        // закрываем сообщение, освобождаем ресурсы
        zmq_msg_close(&req);

        zmq_msg_t reply;
        // инициализируем ответ
        zmq_msg_init(&reply);
        // принимаем ответ
        zmq_msg_rcv(&reply, request, 0);
        printf("Received: hello - %d\n", count);
        // закрываем ответ, освобождаем ресурсы
        zmq_msg_close(&reply);
        count++;
    }

    // закрываем контекст
    zmq_close(request);
    // освобождаем ресурсы
    zmq_ctx_destroy(context);
    return 0;
}
```

Достоинство данной библиотеки в ее высокой производительности из-за использования асинхронного ввода/вывода. Существуют порты OMQ на Python, Ruby, PHP, Java, C/ C++/C#, Erlang, Perl.

Недостатки: требуется написание роутера, для реализации, как мы видели, требуется много кода.

В следующей части статьи будет рассмотрено использование очередей с применением NoSQL-хранилищ и специализированных серверов очередей. **EOF**

1. Календарев А. Очереди как структуры данных. Теория и практика. //«Системный администратор», №5, 2014 г. – С. 46-48 (<http://samsag.ru/archive/article/2691>).
2. Официальный сайт OMQ – <http://zeromq.org>.
3. «ZeroMQ». Глава 1: Приступая к работе – <http://habrahabr.ru/post/198578>.

Ключевые слова: параллельные вычисления, очереди, FIFO.



Визитка

МИХАИЛ УШАКОВ, Уральский федеральный университет, разработчик электронной аппаратуры и программного обеспечения для ядерной гаммарезонансной спектроскопии, um.nix.user@gmail.com

Меньше кода – больше дохода

Часть 4. Разработка front-end-слоя

В предыдущих частях статьи [1], посвященных веб-фреймворку Apache Tapestry, были изучены вопросы, связанные с разработкой back-end-слоя, использованием базовых возможностей и т.д. Поговорим о применении JavaScript и CSS-библиотек (front-end-слой) в Apache Tapestry веб-приложении

Использование CSS-фреймворков

Использование CSS-фреймворков позволяет быстрее создавать HTML-разметку веб-страниц, поддерживать адаптивность (представление HTML на всех форматах: десктопы, планшеты, телефоны) и кроссбраузерность.

Однако у них есть и недостатки, например, в большинстве случаев – тяжеловесность, однотипность дизайнов веб-приложений, полная зависимость дизайна от фреймворка. Тем не менее CSS-фреймворки являются хорошим инструментом для быстрого создания веб-приложений.

Наиболее известными являются Twitter Bootstrap, Foundation, Golden Grid System, Cascade и другие. Пожалуй, наиболее богатый по функциональным возможностям Twitter Bootstrap, поэтому рассмотрим, как с его помощью можно создавать дизайн веб-страниц. Его можно загрузить с [github](https://github.com) или домашней страницы [2].

В предыдущей статье [1] было кратко описано, как использовать компоненты и, в частности, Layout. Layout является шаблоном, общим для веб-страниц (можно иметь несколько Layout для отдельных групп страниц). Поэтому основной задачей будет формирование Layout для веб-страниц.

Файловая структура Twitter Bootstrap представляет собой три каталога: собственно самих каскадных таблиц стилей (.css), опциональные JavaScript-плагины для анимации элементов управления (.js) и папка с ресурсами (.img или .fonts, зависит от версии).

Скопируем эти папки с содержимым в `resources/webapp`. И создадим в `/src/main/java/.../components` класс `BootstrapLayout.java` и файл разметки `BootstrapLayout.tml` данного компонента.

Я рассмотрю простейший пример, в котором не будет использоваться передача параметров из Layout в веб-страницы. В Layout необходимо подключить каскадные таблицы стилей и скрипты, необходимые для использования Bootstrap-фреймворка. Сделать это можно несколькими способами:

1) Использовать поля типа `Asset`, которые отвечают за представление ресурсов:

```
@Inject
@Path("/context:/css/bootstrap.css")
private Asset bootstrapStylesheet;
```

А для загрузки каскадных таблиц стилей на страницу необходимо выполнить загрузку таблиц стилей с помощью объекта типа `JavaScriptSupport`:

```
@Environmental
protected JavaScriptSupport javaScriptSupport;
```

И загрузить Asset-объект:

```
javaScriptSupport.importStylesheet(commonStylesheets);
```

2) Использовать аннотации:

```
@Import(stylesheets = "context:layout/layout.css")
public class Layout
{
    // ...
}
```

3) Подключить стили в .tml-разметке:

```
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8"/>
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
<link href="{context:css/bootstrap.css}" rel="stylesheet" />
<link href="{context:css/bootstrap-responsive.css}" rel="stylesheet" />
<title>Layout present</title>
</head>
```

Пожалуй, наиболее удобным для добавления CSS-файлов без перекомпиляции и смешивания подходов является использование последнего способа (добавление стилей в разметке). При написании Layout-компонента необходимо обозначить место в разметке, куда будет вставлена разметка страницы: `<tbody>`.

Создадим для примера простую веб-страницу с Bootstrap-стилями. Для этого достаточно посмотреть примеры по документации по Bootstrap или можно взять готовый Bootstrap-шаблон.

Использование JavaScript-фреймворков

С JavaScript дело обстоит несколько сложнее, поскольку Tapestry использует стек библиотек: prototype.js, script.aculo.js и собственную библиотеку. Поэтому полностью убрать эти библиотеки будет сложно, если в проекте используются стандартные компоненты с Ajax, встроенная валидация форм и отправка сообщений клиенту. Конечно, можно полностью отказаться от встроенных компонентов Tapestry и оставить на страницы методы для работы с запросами и DTO-объектами (см. в предыдущей статье RestEasy). Tapestry хоть и обладает богатыми возможностями по переопределению стандартных сервисов, однако это не относится к переопределению встроенных JavaScript и CSS. Сервис, выполняющий инициализацию JavaScript-стека, создается на лету, поэтому его нельзя переопределить. По заявлениям в Tapestry 5.4 будет реализована возможность управлять стеком JavaScript-библиотек (на данный момент доступна только бета-версия). Также следует отметить, что Tapestry загружает свои JavaScript-библиотеки только лишь по необходимости, поэтому подход с полным созданием пользовательского интерфейса на явном (так как стандартные компоненты также разворачиваются в чистый HTML и JavaScript) HTML и JavaScript. Для front-end-разработки существует огромное количество библиотек для создания гибкого интерфейса пользователя, например, Dojo или jQuery + jQueryUI, или даже целые JavaScript-фреймворки Backbone, Angular, Knockout. Для небольших веб-приложений можно использовать Tapestry5-jQuery, скачав его с github [3], или добавить библиотеку через maven:

```
<dependencies>
...
<dependency>
<groupId>org.got5</groupId>
<artifactId>tapestry5-jquery</artifactId>
<version>3.3.7</version>
</dependency>
...
</dependencies>

<repositories>
...
<repository>
<id>devlab722-repo</id>
<url>http://nexus.devlab722.net/nexus/content/ </url>
<snapshots>
<enabled>false</enabled>
</snapshots>
</repository>

<repository>
<id>devlab722-snapshot-repo</id>
<url>http://nexus.devlab722.net/nexus/content/ </url>
<releases>
<enabled>false</enabled>
</releases>
</repository>
...
</repositories>
```

В конфигурации можно либо запретить, либо оставить поддержку (для использования стандартных компонентов). Для отключения prototype.js в AppModule.java необходимо добавить следующую строку конфигурации:

```
configuration.add(JQuerySymbolConstants.SUPPRESS_PROTOTYPE, <boolean>
"false");
```

Однако у этого подхода есть недостатки: обновление версий jQuery и jQueryUI будет проходить только с перекомпиляцией проекта, однако, так же как и с CSS, можно сделать смешанный подход и подключать JavaScript-библиотеки в .html-разметке страниц:

```
<script type="text/javascript" <script>
src="(context:/js/jquery-2.0.0.js)" />
```

Скрипты, так же как и CSS, могут быть добавлены через Asset и JavaScriptSupport:

```
javaScriptSupport.importJavaScriptLibrary(jQueryCore);
```

При использовании jQuery совместно с JavaScript-стеком Tapestry могут быть проблемы, так как jQuery использует функцию \$(), которая определена и в prototype.js, следовательно, могут быть проблемы с разрешением имен. В этом случае придется использовать jQuery() вместо \$() (\$ – псевдоним jQuery), а также разрешить конфликты с помощью функции jQuery.noConflict(). Аналогичным образом могут быть добавлены любые JavaScript-библиотеки. Единственным важным моментом является то, что не стоит сочетать встроенные в Tapestry JavaScript-библиотеки с добавляемыми, иначе могут быть проблемы в разрешении имен функций и т.д.

В целом, несмотря на ряд недостатков (недостаточную гибкость конфигурации по отношению встроенных таблиц стилей и стека JavaScript-библиотек), с использованием JavaScript-фреймворков в Tapestry веб-приложении можно построить гибкий и дружелюбный пользовательский интерфейс с минимальными затратами усилий. Также следует отметить, что в версии 5.4 одной из новых фиш будет улучшение поддержки конфигурирования используемых JavaScript-библиотек. **Box**

1. Ушаков М.В. Меньше кода – больше дохода. Часть 1. Создание проекта и конфигурирование. //«Системный администратор», №1-2, 2014 г. – С. 98-101 (<http://samag.ru/archive/article/2620>).
2. Ушаков М.В. Меньше кода – больше дохода. Часть 2. Разработка backend-слоя. //«Системный администратор», №3, 2014 г. – С. 50-55 (<http://samag.ru/archive/article/2644>).
3. Ушаков М.В. Меньше кода – больше дохода. Часть 3. Tapestry для создания Java EE-приложений. //«Системный администратор», №5, 2014 г. – С. 56-60 (<http://samag.ru/archive/article/2694>).
4. Страница для загрузки Twitter Bootstrap – <http://getbootstrap.com/getting-started/#download>.
5. Страница Tapestry5-jQuery – <https://github.com/got5/tapestry5-jquery>.

Ключевые слова: Tapestry, фреймворк, CSS, Twitter Bootstrap.



Визитка

ДМИТРИЙ АНДРИЕНКО, работает в ООО «Ютел»,
программист-ERP, andrienko.dmitry@gmail.com

Бесполезные занятия

Проблемы российского ИТ-образования. Взгляд изнутри

«Хочешь получить корочку? Молчи и не мешай!» — сказал мне преподаватель. Увы! Неквалифицированные педагоги делают неэффективным обучение ИТ-специалистов в наших техникумах и вузах

С началом активного развития автоматизации российский бизнес начал внедрять разные информационные технологии, которые преимущественно импортировались с Запада. Для их адаптации требовались достаточно квалифицированные специалисты со знанием английского языка и опытом работы с вычислительными системами. В бывшем СССР их были единицы, и то они работали больше с отечественными системами, которые значительно отличались от западных аналогов. Данная ситуация вызвала острый дефицит на рынке труда, что давало этим специалистам возможность получать неплохую зарплату. В итоге престиж специальности «программист» в обществе вырос, и на это отреагировала отечественная система образования, включив программистов в пул продаваемых специальностей наряду с юристами, экономистами и нефтяниками.

Предложения на рынке образовательных услуг

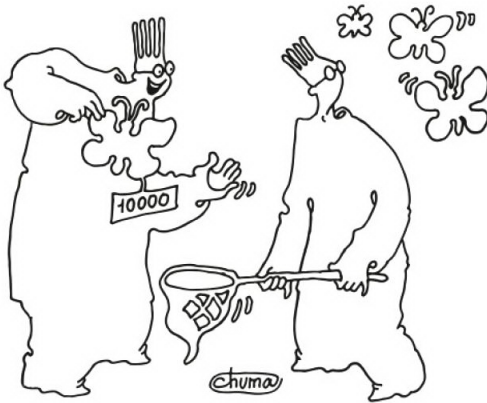
Большинство российских учебных заведений (говорим о высших учебных заведениях и среднепрофессиональных, затрагивать разные курсы и тренинги не будем) открыло у себя кафедры информатики, куда требовались преподаватели, но зарплаты работников образования настолько малы, что пригласить туда хорошего специалиста просто невозможно. И руководство вузов напичкало учебные программы по этим специальностям математикой, физикой и прочим, а на полезные в работе будущему программисту спецпредметы поставили тех же преподавателей математики и информатики, которых немного переучили по разным переведенным зарубежным книгам по программированию. Теперь данная бизнес-конструкция может работать и производить «отличных» ИТ-специалистов, принося руководству университетов прибыль и удовлетворяя потребности абитуриентов в дипломах.

Из предлагаемых на рынке специальностей можно выделить наиболее популярные: «Автоматизированные системы управления», «Программное обеспечение вычислительной техники и автоматизированных систем», «Электронно-вычислительные машины (ЭВМ)», «Информационная безопасность», «Информатика в экономике» и т.п., названия варьируются из года в год. Рынок вынуждает подбирать более современ-

ные названия, например, «Управление информационными системами», не удивлюсь, если скоро появится специальность «Управленческий менеджмент информационных систем», уверен, что от абитуриентов не будет отбоя. Кстати, во время последней моей сессии (я учусь заочно по специальности «Прикладная информатика в экономике») я попал на предмет «Управление информационными ресурсами», где преподаватель (из группы перечисленных учителей информатики) рассказывал о должности СЮ (заместитель генерального директора по информационным технологиям, директор по ИТ). Преподаватель говорил: «Вы будущие СЮ, менеджеры высшего звена и можете по окончании вуза претендовать на эту должность». Конечно, для меня как специалиста, проработавшего в разных компаниях и на различных ИТ-должностях, это было шоком, но пришлось промолчать, ибо если открыто возразить преподавателю, то можно подвергнуться очень жесткой прессовке в виде заниженных оценок, многократных перепечаток курсовых работ, пересдач зачетов и экзаменов. Однажды один из преподавателей во время жаркого спора в начале моего обучения сказал мне: «Ты хочешь получить корочку? Так молчи и не мешай мне работать!» — а как по мне, так это звучит: «Не пали нас и не мешай рубить бабло...»

Содержание и качество образовательного процесса

Процесс моего ИТ-обучения начался в возрасте 15 лет, когда я поступил в техникум (недоколледж) по специальности «Программное обеспечение вычислительной техники и автоматизированных систем». У меня был интерес к информационным технологиям, да и уроки информатики очень нравились, поэтому мой выбор был осознан, но в голове был туман — какие там представления, просто ребенок. В техникумах картина намного печальнее, чем в вузах: зарплата преподавателя мизерная, да и уроки информатики и не амбициозные. Правда, есть и работающие за идею, но таких — единицы. Мне попалась неплохая преподавательница по программированию, бывший советский программист, которая преподавала Pascal и Delphi, Visual C++. Проблема преподавания старых и малопопулярных языков



Полезный инструмент мотивации — денежная премия преподавателю за устройство выпускника по специальности

программирования сохраняется, она же касается Pascal и Delphi 7, и все потому, что преподаватель, когда-то освоив один язык, далее не желает переучиваться, не обращая внимания на тенденции ИТ-рынка. Был неплохой преподаватель баз данных, но тут тоже беда: FirebirdSQL + Delphi, что также показывает нежелание переучиваться, очень мало времени уделялось системному администрированию. Как и в институте, много предметов, направленных на общее развитие учащегося. Техникумы и колледжи — это промежуточное звено, ибо с таким образованием можно претендовать только на позиции помощников или малооплачиваемые должности, все равно придется идти и получать высшее.

Получение высшего образования идет по похожей схеме: много общих дисциплин и мало внимания уделяется профессиональным предметам, а практике — еще меньше. Преподаватели значительно более высокого уровня, но основная часть их знаний — теория, практика минимальная и складывается из рассказов заочников, работающих по специальности, а также дипломников, внедряющих какой-нибудь малозначимый проект на «производстве». Большинство преподавателей — это бывшие студенты, которые сразу после окончания обучения поступили в аспирантуру, они не имеют представления о состоянии рынка труда и потребностях работодателя. Но зато преисполнены чувством собственной значимости. Например, один из них умудрился завалять на экзамене специалиста с 12-летним опытом в ИТ по предмету «Операционные системы». Специалист имел сертификат и отличался глубокими знаниями, а главное, обширным опытом, но преподаватель только после нескольких передач поставил ему тройку.

Еще один феномен — это получение высоких оценок людьми не из ИТ. Допустим, охранник или воспитатель детского сада, регулярно посещающий занятия, вовремя списывая, смотря в рот преподавателям и покупая работы у «решал», умудряется получать намного более высокие оценки, чем ИТ-специалисты.

Имеются, конечно, и преподаватели, которые где-то поработали или работают, люди из данной сферы, что-то повидавшие. Обычно они придерживаются следующего стиля:

«Да, я понимаю, что вы специалисты, но я хотел бы рассказать вам о своей дисциплине подробнее, посвятив вас в детали, ответить на вопросы и поделиться опытом» — такой подход гораздо более полезный.

Предложения по совершенствованию

Конечно, первоочередным будет увеличение заработной платы преподавателя, чтобы она превышала зарплату, например, продавца-консультанта. Не следует допускать к такого рода деятельности человека, не проработавшего по специальности «на производстве» менее трех лет, — это позволит убрать с кафедры людей инфантильных и не понимающих, что они делают и зачем. Немаловажным будет уменьшение общих дисциплин и заострение внимания на профессиональных, увеличение количества практических часов. Еще одним полезным инструментом мотивации станет денежная премия преподавателю за устройство выпускника по специальности, например, руководителю дипломного проекта (он же руководитель производственной практики). Преподаватели будут заинтересованы устанавливать контакт с работодателем и передавать им студентов.

Обучение в российском высшем учебном заведении по направлению ИТ, по моему мнению, малополезно в профессиональном плане, возможно, оно и развивает человека в общем, но не сделает из него специалиста. Высшее образование лишь увеличит шанс получить стажировку или устроиться на начальные позиции новичкам в мире ИТ. Люди, которые уже работают в ИТ, для повышения своей конкурентоспособности или получения новых должностей могут получить образование по заочной форме обучения, но столкнуться со спецификой, описанной в статье, им тоже придется, хотя и в меньшей степени. Обучаться в среднем профессиональном учебном заведении на ИТ — на мой взгляд, бесполезное занятие. **БОБ**

Ключевые слова: ИТ, образование, обучение, подготовка специалистов.



Визитка

ИГОРЬ ШТОМПЕЛЬ, инженер, системный администратор.
Сфера профессиональных интересов — свободное программное обеспечение, keepercoder@gmail.com

Программирование на Java

Программирование — интереснейшая и захватывающая область ИТ. Но, чтобы войти в эту область, нужно выбрать и изучить язык программирования. Согласно TIOBE Index второй по популярности язык — Java. Для тех, кто желает познакомиться или углубить свои знания, мы предлагаем перечень интересных курсов

Программирование на Java

- > Автор: В. Вязовик
- > Учебное заведение: ИНТУИТ (intuit.ru) при поддержке Sun Microsystems
- > Специальность: Программист
- > Лекции: 16 + 2 практикума
- > Материалы: Текст, видео
- > Уровень сложности: Специалисты
- > Сертификат: Да
- > Источник: <http://www.intuit.ru/studies/courses/16/16/info>

Предлагаемый курс носит вводный характер, является введением в язык программирования Java и основы объектно-ориентированного программирования (ООП). Материалы курса начинаются с истории и развития языка, особенностей платформы Java. Продолжают курс лекции по основам ООП (методология процедурно-ориентированного программирования; методология ООП; объекты; классы; типы отношений между классами; достоинства и недостатки ООП), лексике Java (кодировка; анализ программы; виды лексем: идентификаторы, ключевые слова, литералы; операторы; операции), типам данных (переменные; примитивные и ссылочные типы данных и др.). Дальнейшие лекции расширяют кругозор и посвящены изучению: имен и пакетов, объявлений классов, преобразованию типов, объектной модели в Java, массивов, операторов и структуры кода, исключений, потоков выполнения, синхронизации, введения в сетевые протоколы. Кроме того, отдельные лекции курса посвящены пакетам: java.awt (дерево компонентов; обработка пользовательских событий; апплеты; менеджеры компоновки), java.lang (Object; Class; классы-обертки; Math; строки; системные классы; потоки исполнения; исключения), java.util (работа с датами и временем; интерфейс Observer и класс Observable; коллекции; класс Properties; интерфейс Comparator; класс Arrays; класс StringTokenizer; класс BitSet; класс Random; локализация), java.io (система ввода/вывода; потоки данных (stream); сериализация объектов (serialization); классы Reader и Writer и их наследники; класс StreamTokenizer; работа с файловой системой). Курс позволяет получить базовые знания о языке программирования Java, которые станут прочным фундаментом для дальнейшего его освоения.

В свою очередь, дополнение курса в виде двух практикумов, посвященных разработке приложений, в том числе веб-приложений, в среде Eclipse, дают первоначальное понятие об использовании среды разработки.

Java

- > Автор: А. Владыкин
- > Учебное заведение: Проект Лекториум (lektorium.tv)
- > Специальность: Программист
- > Лекции: 12
- > Материалы: Видео
- > Уровень сложности: Специалисты
- > Сертификат: Нет
- > Источник: <http://www.lektorium.tv/course/22896>

Еще одним хорошим введением в программирование на языке Java может служить данный курс, который стал достаточно популярным на портале проекта Лекториум. Он позволит закрепить базовые знания, полученные при изучении курса, рассмотренного выше. Лекции начинаются с рассмотрения истории и эволюции языка Java, его особенностей и экосистемы, стандартных инструментов JDK. После этого автор предлагает ознакомиться с типами данных (примитивными, ссылочными, boolean, целочисленными, вещественными), а также явным и неявным преобразованием типов, классами-обертками, классами Math, BigInteger и BigDecimal. Третья лекция посвящена массивам (объявление, создание, инициализация, индексация, одномерные и многомерные, представление в памяти, java.util.Arrays) и строкам (строковые литералы, операции со строками, java.lang.StringBuilder, поддержка различных кодировок, регулярные выражения). Последующие лекции дают представление об основах ООП (объявление класса; использование класса; наследование и др.), об управляющих конструкциях и исключениях (условные операторы if и switch; циклы for, while и do; операторы break и continue; метки; типы исключений и др.), о стандартной библиотеке Java (java.io; java.nio.file; доступе к файловой системе; java.util; java.lang.reflect и др.), о разработке многопоточных приложений (общие сведения о параллелизме;

управление потоками; синхронизация; модель памяти и др.). В заключение предлагается познакомиться с модульным тестированием (JUnit, Mockito, Java Logging API) и разработкой сетевых приложений (сокеты, URI и URL, библиотека Netty).

Углубленное программирование на Java

- > Авторы: В. Чибриков, А. Акбашев, Е. Шубин
- > Учебное заведение: Проект Технопарк (проект Mail.Ru Group и МГТУ им. Н.Э. Баумана, <https://tech-mail.ru>)
- > Специальность: Программист
- > Лекции: 9
- > Материалы: Видео
- > Уровень сложности: Специалисты
- > Сертификат: Нет
- > Источник: <https://tech-mail.ru/video/learn>

Предлагаемый курс поможет расширить представление о языке программирования Java, предлагаемый в рамках проекта Технопарк (совместное детище Mail.Ru Group и МГТУ им. Н.Э. Баумана). Для успешного прохождения курса необходима предварительная подготовка: знание языка программирования Java на базовом уровне и опыт работы с ним. Первая лекция является введением и дает представление об истории и особенностях Java, платформе Java и ее редакциях, сборщике мусора, запуске Java-приложений, наследовании в Java. Здесь же сравниваются Java и C++. Следующая лекция посвящена многопоточности (Reflection, Class Object и Class Class, классы-обертки простых типов, типизация, обзор коллекций Java, Class Thread, sleep(), interrupt(), join(), потоки с точки зрения процессора, прерывания, Java memory model, синхронизация, мьютексы, семафоры, проблемы многопоточного доступа к данным, пример неатомарности операции ++, примеры многопоточных приложений и др.). Уделяется внимание взаимодействию потоков (описание проблемы взаимодействия потоков, взаимодействие потоков Frontend и AccountService, плюсы решения задачи через два потока, способы взаимодействия потоков через сообщения, примеры кода и алгоритм работы и др.); игровой механике (проблема циклических зависимостей и ее решение через интерфейсы, архитектура сервера, разбор модулей сервера и их взаимодействие, игровая механика, диаграмма взаимодействия модулей игровой механики и фронтенда, репликация и др.); тестированию и нагрузке (виды тестирования: Unit-тестирование, функциональное, нагрузочное; практики тестирования, инструменты: Junit, Selenium; сборщик мусора: виды, параметры, принципиальное устройство и др.); вводу/выводу (события, анонимные классы, работа со временем и датой, подписка на время, I/O Streams, потоки байт и потоки символов, дерево наследования потоков, файловые потоки, безопасность закрытия потоков, VFS, примеры кода и др.); ресурсной системе (сериализация и десериализация, SAX и DOM парсеры, восстановление объекта по XML-файлу, ресурсы, использование ReflectionHelper для десериализации ресурсов и др.); работе с базами данных (шаблон работы с базой, JDBC API, модуль для работы с базой данных и др.); ORM (Object Relational Mapping, объектно-реляционное отображение; Hibernate и др.). Таким образом, данный курс позволяет разработчикам углубить знание о языке программирования Java и подняться выше базового уровня в своей профессиональной компетенции.

Язык программирования Java и среда NetBeans

- > Автор: В. Монахов
- > Учебное заведение: ИНТУИТ (intuit.ru) при поддержке Sun Microsystems
- > Специальность: Программист
- > Лекции: 12
- > Материалы: Текст
- > Уровень сложности: Специалисты
- > Сертификат: Да
- > Источник: <http://www.intuit.ru/studies/courses/569/425/info>

В этом курсе подробно рассматривается среда NetBeans. Автор начинает изложение материала с общих вопросов (Java и другие языки программирования, виртуальная Java-машина, байт-код, JIT-компиляция, алфавит языка Java, специальные символы и т.п.), а затем переходит к проблемам объектно-ориентированного проектирования и среде NetBeans (процедурное и ООП, проекты NetBeans, создание в NetBeans простейшего приложения Java, компиляция файлов проекта и запуск приложения, структура проекта NetBeans, создание в NetBeans приложения Java с графическим интерфейсом, документирование исходного кода в Java, технологии Java и .Net.). Последующие лекции посвящены непосредственно языку Java: примитивным типам данных, операторам для работы с примитивными типами данных, работе с числами, управляющим конструкциями, объектно-программированию, важнейшим объектным типам, наследованию, интерфейсам, композициям, дополнительным элементам объектного программирования, сетевому программированию, встроенным классам. Последняя лекция возвращает к работе в среде NetBeans в рамках рассмотрения компонентного программирования (компонентная архитектура NetBeans, мастер создания компонента в NetBeans, пример создания компонента, добавление в компонент новых свойств и событий и др.). Таким образом, особенностью курса является не только предоставление базовых знаний о работе в среде NetBeans, но и о языке программирования Java.

Использование и расширение Eclipse

- > Автор: А. Иванов
- > Учебное заведение: ИНТУИТ (intuit.ru)
- > Специальность: Программист
- > Лекции: 5
- > Материалы: Видео
- > Уровень сложности: Специалисты
- > Сертификат: Да
- > Источник: <http://www.intuit.ru/studies/courses/62/62/info>

Еще одна очень популярная среда среди Java-программистов – Eclipse, которой и посвящен рассматриваемый курс. Автор знакомит с самой средой Eclipse (что она собой представляет как IDE и как сообщество). Следующая лекция посвящена основам разработки (запуск и настройка Eclipse, создание и открытие проекта программы, сборка и запуск программы и др.). Продолжают материал сведения об отладчике среды Eclipse (запуск отладчика, интерфейс и т.п.). Заключительные лекции посвящены подключению и использованию Subversion (системы контроля версий) и Atlassian Jira (системы учета задач). Курс полностью соответствует заявленным задачам и позволяет в указанных рамках освоить использование среды Eclipse. **PDF**



Визитка

ВЛАДИМИР ИВАНОВ, специалист по информационной безопасности. Увлекается психологией, историей, философией

Безопасное трудоустройство

Часть 5. Где лучше работать российским ИТ-специалистам

В статье пойдет речь о возможных перспективах при трудоустройстве в различные компании, а также будут даны рекомендации для рассмотрения тех или иных «заманчивых» предложений о работе

В давние времена люди, замученные тяжелыми повинностями и несправедливой властью, мечтали о далекой стране Беловодье, где царит Свобода. Где нет злых царей, князей, помещиков, капиталистов, чиновников и прочих, угнетавших собственный народ. Где земля плодородна, а жители добры и приветливы.

Отголоски этих давних легенд можно найти в представлении современных ИТ-специалистов о том, что есть такие чудесные компании, где к инженерному персоналу относятся «как к людям», где нет пиратского программного обеспечения, «серых» зарплат, неоплачиваемых переработок и прочих прелестей среднестатистического российского бизнеса.

Мы рассмотрим несколько популярных мифов, которые до сих пор гуляют по просторам Интернета и оседают в головах российских айтишников.

Нужно идти работать в «зарабатывающее» подразделение

То есть работать в тех компаниях, чей доход напрямую связан с ИТ-технологиями. Основан данный миф на простом здравом смысле, никто в трезвом уме не будет «душить курицу, которая несет золотые яйца». Если бизнес зарабатывает, используя труд своего инженерного персонала, то этому персоналу нужно как минимум обеспечить достойное вознаграждение и приемлемые условия труда.

Увы, в российском бизнесе все не так радужно. Сколько бы ни зарабатывали инженерные подразделения, они, так или иначе, находятся в зависимости от управленцев и других представителей менеджмента. В итоге ИТ-специалисты продолжают трудиться за небольшую заработную плату, при этом постоянно ломая голову, как выполнить свою работу в условиях скудного финансирования, а все лавры и бонусы достаются либо руководству более высокого ранга, а то и вовсе подразделениям сугубо экономического профиля.

Надо отметить, что те, кто распространяет сказки о «зарабатывающих» подразделениях, не потрудились соотносить данные мифы с конкретными направлениями бизнеса.

Например, ИТ-служба в банке – с одной стороны, современный банк не может выжить без ИТ, с другой стороны, айтишники не принимают деньги у вкладчиков и решение о выдаче кредита.

Пока в головах ИТ-специалистов и других наемных днюков не установится понимание того, что нет и не может быть никакого деления на «зарабатывающие» и «незарабатывающие» подразделения, никакого результата в плане повышения материальных благ и улучшения условий труда не будет. Чтобы выжить и нормально развиваться, любой бизнес должен работать слаженно. При этом все подразделения, службы и отдельные сотрудники неразрывно связаны между собой.

Какой бы специфичной работой ни занимались представители инженерных профессий, они всегда будут в роли ездовой собаки. А менеджеры всегда найдут способ получать побольше и работать поменьше. Потому что инженеры работают с техникой, а менеджеры – с людьми, а именно люди зарабатывают и приносят деньги.

Поэтому нужно не бежать устраиваться на работу в «зарабатывающее» подразделение, а выбирать хорошие условия труда и высокую заработную плату.

Нужно стремиться работать в большой компании

Основное заблуждение кроется в предпосылке, что все текущие проблемы связаны с финансовыми трудностями бизнеса, а в большой богатой компании таких проблем нет. Якобы в большой компании лучше условия труда, больше платят и к ИТ-службе в целом относятся с гораздо большей симпатией.

На самом деле все может быть с точностью наоборот. Многие крупные компании являют собой полное сосредоточение хаоса, где никто ничего не знает и никто ни за что не отвечает, а во всем виноват «стрелочник», в данном случае ИТ-служба. Или же они организованы по принципу антиутопии, где царствуют тотальный контроль и полностью механистический подход к управлению персоналом.

Цель превращения сотрудников в «винтики» и «шестеренки» гигантского бизнес-механизма – вот такая супер-



Развенчаем популярные мифы, которые гуляют по просторам Интернета и оседают в головах российских айтишников

дея часто царит в мозгах многих «эффективных» менеджеров.

Карьерный рост в таких условиях может быть весьма ограничен. Молодой человек, устраивающийся на работу в качестве сотрудника службы техподдержки, вполне может уволиться через несколько лет таким же «мальчиком-саппортом». Причина в том, что в последнее время компании демонстрируют полное нежелание вкладывать деньги в развитие собственных сотрудников, предпочитая искать на рынке готовых специалистов.

Перспектива вырасти от энкейщика до ИТ-директора – это миф. ИТ-директора будут искать на стороне, среди людей, уже имевших опыт работы в данной должности, а энкейщика выгонят, как только он нахватается отрывочных знаний и начнет претендовать на что-то большее, чем настройка рабочих станций.

Популярная для крупных компаний роль «человековинтика» большой перспективы не имеет, зато «загнанных лошадей пристреливают».

Отчасти это произошло из-за кризиса 2008-2009 годов, а отчасти просто потому, что в России царит атмосфера «обезьяньего бизнеса», работающего по принципу «сорвал и на дерево». Остается только пожалеть тех несчастных, кто вложил в такую компанию много времени и сил в расчете на хоть какой-то карьерный рост и в итоге оказался у разбитого корыта.

Примечание. Как раз именно в маленькой компании есть небольшие шансы пройти путь от «помощника системного администратора» до «системного администратора» и до «начальника ИТ-отдела». И хотя это случается совсем не часто, но такие случаи имели место быть. К сожалению, последняя должность из этой триады является своего рода «карьерным потолком». Дальше нужно будет либо менять профессию на «управленческую», либо довольствоваться тем, что имеешь.

Еще один важный момент. В крупных компаниях затруднены вертикальные коммуникации. Если в небольшой или даже средней компании системный администратор здоровается за руку с генеральным директором и может

при случае обратиться к нему с личной просьбой, то в крупной компании такой вариант выглядит нереальным. Зачастую проще попасть на прием к министру, чем к директору филиала большого предприятия.

В качестве замены общения с руководством наемным работникам всех уровней предлагают всевозможные «аттестации персонала», «тимбилдинги» и прочие мошеннические схемы манипуляции персоналом (см. предыдущие статьи из серии «Безопасное трудоустройство» [1-4]). Соответственно при такой схеме о каких-либо перспективах повышения зарплаты говорить не приходится, а вот шанс, что по итогам «аттестации» размер вознаграждения будет снижен, очень даже возможен.

Данные проблемы не зависят ни от профиля компании, ни от рода деятельности, ни от структуры капитала. Никакие нефтяные компании, крупные банки, госкорпорации не могут гарантировать отсутствие проблем, как то неоплаченные переработки, хамское отношение со стороны руководства и рядового персонала, отсутствие карьерного роста, навязывание дресс-кода и так далее.

Основная рекомендация для соискателей: «Не смотрите на размер компании, а оцените условия, которые вам предлагают. Скорее все эти условия останутся либо неизменными за все время вашей работы в компании, либо их сделают хуже, например, под предлогом очередного экономического кризиса».

Нужно держаться поближе к деньгам

Такое мнение популярно среди людей, на самом деле далеких от каких-либо экономических познаний. Мол, если мимо проходят какие-то там финансовые потоки, то и вполне может что-то перепасть.

Неизвестно, кто и когда придумал эту глупость. Дело в том, что чем больше оборот капитала на предприятии, чем выше требования к безопасности. В итоге люди оказываются в ловушке собственного самообмана, когда масса сил и жизненных ресурсов уходит на выполнение бесчисленных правил, регламентов и нормативов в обмен на весьма посредственное денежное вознаграждение. При этом

всегда есть риск быть уволенным или даже пойти под суд за малейшую ошибку.

Напротив, ИТ-специалисты из «небогатых» компаний могут позволить себе с большим размахом наслаждаться жизнью, время от времени занимаясь не только работой с непрекращающимся потоком заявок, но и «личными» проектами для самообразования и повышения квалификации, а также просто для заработка.

Нужно не бежать устраиваться на работу в «зарабатывающее» подразделение, а выбирать хорошие условия труда и высокую заработную плату

Совет простой: сопоставьте, соответствует ли размер обещанного вознаграждения тому количеству хлопот и ограничений, которое придется перенести из-за такой вот «близости к финансовым потокам».

Имя компании говорит само за себя

Это предположение довольно спорно. Конечно, «красивое» резюме (с точки зрения работника службы персонала) дает больше шансов получить интересную и высокооплачиваемую работу. В то же время привлекательность предыдущего опыта работы каждый оценивает по-разному. Например, служба персонала может оценивать резюме по громкости брендов, а руководитель ИТ-службы (которая в принципе и является в данном случае прямым заказчиком) гораздо большее значение придает уровню подготовки и личным качествам будущего подчиненного.

«Лучше быть первым в деревне, чем вторым в Риме». Эта мудрость, похоже, никогда не потеряет актуальность. Простой пример: на должность ИТ-директора с гораздо большей вероятностью возьмут ИТ-директора из менее крупной компании, чем хорошего менеджера среднего звена, своего сотрудника.

Помимо всего прочего, в известных брендах чаще всего присутствуют все негативные факторы крупных компаний. Поэтому стать «человековинтиком» здесь гораздо проще, чем на менее «заметном» предприятии. Например, в рекламных роликах некоторых известных компаний демонстрируются лозунги вроде «сисадмин у нас – первый человек», а на деле практикуется классическая «палочная дисциплина» по отношению к «обслуживающему персоналу», которым считают и системных администраторов.

Подытожим все вышесказанное: «Я работал в...» – это звучит здорово, пока не спросят: «А что ты там делал?»

Но это еще не все. Часто бывает, что предложение по зарплате в известных компаниях оказывается значительно скромнее, чем в менее раскрученных. Имеет место быть такая классическая манипуляция: «Вы понимаете, за все надо платить, в том числе и за престиж...» Или говорят: «Зато после работы в нашей компании вас с удовольствием возьмут куда угодно». Но куда угодно не возь-

мут, и совсем не факт, что вообще получится быстро найти работу, этот вопрос мы уже рассмотрели выше. А платить собственной упущенной выгодой за престиж работодателя бессмысленно.

Один из поведенческих приемов для таких ситуаций – в ответ на бодрое заявление HR-менеджера на тему «вам предоставляется уникальный шанс поработать в известной компании» сказать что-то вроде: «Это для меня существенной роли не играет, я работаю хорошо в любом случае, независимо от известности компании». Если под маской известного бренда скрывается «соковыжиматель», HR постарается от вас поскорее избавиться, потому что попытка манипулирования провалилась.

Западные компании лучше российских?

И да, и нет. Все зависит от конкретных параметров, по которым определяются преимущества.

Во-первых, под вывеской «западная» или «зарубежная компания» могут находиться предприятия, абсолютно различные по своей сути. В одних бок о бок работаешь с экспатами – выходцами из развитых капиталистических стран. При этом иностранцы работают на различных должностях: они не только топ-менеджеры, а и руководители среднего звена, консультанты и так далее. Соответственно уровень, методы управления и корпоративная культура будут сильно приближены к западным стандартам.

Другое дело, если компания вроде западная, а руководство и практически весь персонал – российские граждане, за небольшим исключением. В этом случае работа в данной организации ничем не отличается от классической российской «конторы», иногда даже с худшими условиями. Например, во время небылой жары в Москве летом 2010 года российское руководство одной из таких «западных» компаний попыталось сократить рабочий день из-за высокой температуры, на что иностранные топ-менеджеры потребовали отменить распоряжение и оставить изнывающих от жары людей на рабочих местах.

Отличить эти два варианта довольно легко. Если вакансия связана с непосредственной работой с экспатами и иностранцев в данном месте довольно много, от соискателя потребуют хорошее знание английского языка. Часто соискателей просят пройти дополнительное интервью на требуемом иностранном языке. Обычно необходимо знание английского языка на уровне Intermediate [5], но может потребоваться владение французским, немецким, испанским или другим языком ведущих мировых держав.

Что же касается «иностранных» компаний с русским менеджментом, как правило, знание языка или не требуется вовсе, или работодатель готов закрыть глаза на недостаточный уровень языковой подготовки кандидата.

Но есть некоторые черты, которые выгодно отличают западные фирмы от российских.

Например, в компаниях с иностранным капиталом более трепетно относятся к вопросам лицензирования программного обеспечения. Отчасти из-за традиционного западного уважения к частной собственности, отчасти из-за нежелания идти на конфликт с местными властями. Если системный администратор в российской компании, увидев знакомую надпись «продукт распространяется по лицензии GPL...», радостно заявляет руководству, что это «бесплатно», то бо-

лее чем вероятно, что в западной фирме его настойчиво спросят, действительно ли этот продукт распространяется бесплатно, нет ли ограничений по поводу коммерческого использования, нет ли требований по поводу обязательного приобретения платной поддержки и уточнялись ли все эти вопросы у правообладателя. В большинстве российских компаний на такие «мелочи» все еще привыкли закрывать глаза.

Еще один интересный момент – требование к разделению труда. Например, системный инженер, порывающий систему на своем компьютере, будет, мягко говоря, неправильно понят. Для простых функций есть служба техподдержки. Задача инженера при таком подходе сводится к работе над сложными решениями. И никакие аргументы вроде «даже генерал сам чистит свое табельное оружие» не находят отклика. Все настроено на то, чтобы выкачать как можно больше денег из наемного персонала.

Рекомендации для соискателей. Если действительно есть сильное желание поработать в западной компании, например, в целях дальнейшей иммиграции в развитую страну, то нужно обратить внимание на языковые требования. В противном случае надо очень придирчиво изучить то, что вам предлагают. Иначе есть риск попасть в ситуацию, когда от работников требуют производительности труда на уровне западных стандартов, при этом заработная плата, условия труда и материальное обеспечение бизнес-процессов будут в рамках худших традиций российского бизнеса.

Несколько практических советов

Как видим, когда дело касается реальности, иллюзии о чудесных компаниях, в которых местные айтишники живут, как в сказке, куда-то улетучиваются. Поневоле задумаешься о том, стоит ли искать «компанию своей мечты» или изучить другие, более «приземленные» варианты.

В первую очередь не надо закидываться в рамках одной-двух профессий. Если нет работы с приемлемыми условиями по своей специальности, стоит задуматься о смене профессии. Нужно уметь находить у себя нереализованные способности и уметь ими пользоваться. Хороший бухгалтер, обслуживающий две-три фирмы на дому, может зарабатывать не меньше, чем, например, инженер по оборудованию Cisco с сертификатом CCNP. Надо просто почаще прислушиваться к себе и подходить с умом к выбору новой работы.

Не следует доверять никаким «хвалебным» отзывам. При просмотре любой публикации о компании следует искать информацию «между строк». Если отзыв слишком оптимистичный, слащавый, напыщенный, если в нем есть слишком много обещаний, скорее всего это дезинформация с целью получить «канарейку за копейку», и чтоб пела и не ела». Помните, что любые интервью, любые «аналитические обзоры», любые «статистические отчеты» и любая другая информация из разряда «для соискателей» – это всего лишь сладкая ложь для усиления бдительности. Чем больше компания себя хвалит, тем критичнее нужно отнестись к ее предложению о работе. Можно сказать, даже жестче. Чем больше представители той или иной компании демонстрируют позитивное отношение к ИТ-службе, тем опаснее туда трудоустраиваться. Скорее всего это жесткий «соковыжиматель», просто прикрытый «позитивным информационным фоном».

Зачастую суровая реальность именитых брендов, окружившая новичков после трудоустройства, способна вогнать в депрессию даже самых стойких.

На самом деле предпосылки такого незавидного положения российских ИТ-специалистов исходят от них самих. Малейшая консолидация усилий в стремлении обезопасить себя и своих коллег от произвола работодателей дает весьма положительный эффект.

Изучение информационных источников в сочетании с тотальным недоверием к любой хвалебной информации, внимательность к мелочам и непредвзятое отношение – пока что это единственные верные союзники соискателя

К сожалению, в этом плане среди российских айтишников пока что полный разброд и шатание. В отличие, например, от авиадиспетчеров или шахтеров представители инженерных профессий так и не научились договариваться между собой и выдвигать совместные требования. (Попытки осмыслить данную проблему время от времени всплывают в блогах [6].) Поэтому при трудоустройстве каждый кандидат может рассчитывать исключительно на свои силы.

Тщательное изучение всех информационных источников, особенно «черных списков работодателей», в сочетании с тотальным недоверием к любой хвалебной информации, внимательность к мелочам и непредвзятое отношение – пока что это единственные верные союзники соискателя.

Надеюсь, что рекомендации, данные в статье, помогут читателям найти действительно достойную работу. В следующей статье мы рассмотрим наиболее популярные типы компаний для трудоустройства российских ИТ-специалистов. **БОФ**

1. Иванов В. Безопасное трудоустройство. На что обратить внимание при просмотре вакансий. //«Системный администратор», №3, 2013 г. – С. 78-82
2. Иванов В. Безопасное трудоустройство. Часть 2. Собеседование и все, что с ним связано. //«Системный администратор», №5, 2013 г. – С. 76-80.
3. Иванов В. Безопасное трудоустройство. Часть 3. Что помогает или мешает работать и зарабатывать. //«Системный администратор», № 1-2, 2014 г. – С. 120-123 (<http://samag.ru/archive/article/2625>).
4. Иванов В. Безопасное трудоустройство. Часть 4. Негативные факторы. //«Системный администратор», № 4, 2014 г. – С. 86-89.
5. Уровни владения английским языком – <http://www.eduwow.ru/levels.htm>.
6. Отсутствие профсоюза айтишников как зеркало судьбы одиочек – <http://www.sql.ru/blogs/taper/938>.

Вакансия: тестировщик ПО

Важным процессом, позволяющим проверить качество кода, является тестирование. Эта работа так и называется «тестировщики ПО», профессия очень актуальна и востребована на ИТ-рынке. Мы спросили у представителей компаний, каким требованиям должен удовлетворять кандидат-тестировщик

1. Какими знаниями и навыками должен обладать тестировщик ПО?
2. Каков инструментарий тестировщика ПО?
3. Каковы требования компании к уровню образования потенциальных сотрудников?
4. Какие требования предъявляются к опыту работы?
5. Есть ли особые требования, которые обусловлены спецификой деятельности компании?

Тимофей Сургученко,

начальник департамента контроля качества в Parallels

1 Для нас самое важное – это общая компьютерная грамотность, аккуратность и коммуникабельность. Без коммуникабельности никуда, тестировщику вообще постоянно приходится что-то у кого-то узнавать, с кем-то о чем-то договариваться, обосновывать неправильность того или иного поведения системы и т.д. Без аккуратности, само собой, тоже в тестировании делать нечего. Кстати, самый простой способ НЕ попасть к нам на собеседование – это сделать опечатку (подчеркиваю – именно опечатку, а не орфографическую или грамматическую ошибку) в резюме, для тестировщика это недопустимо. Без общей компьютерной грамотности тоже никуда – спектр задач, решаемых тестировщиками, очень широк, и, если он будет постоянно лазить в Google, чтобы понять, что такое ошибка 404 или чем отличается IP-адрес от MAC-адреса, его продуктивность будет сильно ниже желаемой.

2 Голова и руки – вот основной инструментарий тестировщика ПО в нашей компании. А помогают ему в его нелегкой работе такие инструменты, как Atlassian Jira (где у нас построен весь технологический процесс работы с дефектами, задачами и функционалом), Atlassian Confluence (где мы храним требования, а также всю внутреннюю базу знаний), TestLink (переписанный нами чуть менее чем полностью – там мы храним и выполняем тестовые сценарии), а также огромное количество самописных сервисов и тулзов (от редакции: tools-инструменты) для автоматизированного развертывания тестовых инфраструктур, написания и прогона автоматизированных тестов, аналитики тестовых результатов и покрытия

продуктов тестами, сбора метрик по проектам и людям и много каких еще задач. Современные подходы к QA постоянно эволюционируют, как и наши инструменты, поэтому мы постоянно что-то дописываем, докручиваем и пробуем новые. Не думаю, что когда-либо мы сможем заявить, что текущим набором тулзов мы удовлетворены.

3 Специфических требований к образованию кандидатов у нас нет. Конечно же, как и любая другая компания, мы рады видеть у себя на собеседованиях выпускников ведущих технических вузов, таких как МФТИ, МГУ, МИФИ, Бауманка и т.д. Но на собеседование к нам может попасть выпускник или студент любого вуза, будь то технарь или гуманитарий, известный вуз или неизвестный техникум. У нас в штате есть люди даже и вовсе без высшего образования. А далее все зависит от самого кандидата, как он покажет себя на собеседовании. Ради интереса поднял резюме нанятых к нам в отдел за последний год – там выпускники Мехмата МГУ, МФТИ, МИЭТа, МАДИ, Бауманки, МГСУ, МФПА, НГУ, БГУ, Кемеровского государственного университета, Хакасского государственного университета и т.д.

4 В идеале нам, конечно же, хотелось бы видеть у себя людей с 5+ годами опыта работы по специальности в одной из ведущих продуктовых ИТ-компаний России и мира. Увы, в реальности таких кандидатов на рынке труда почти нет, или за ними ведется настоящая охота и идет бескомпромиссная борьба между потенциальными работодателями. А люди нужны. Поэтому к себе в отдел мы часто принимаем новичков совсем без опыта работы по специальности. И дальше растим их сами. В основном такие люди имеют

солидный опыт работы в другой интересной нам специальности (например, в системном администрировании или программировании) и какие-то достижения, а также отвечают общим требованиям.

Кроме того, мы практически не нанимаем людей со стороны на ведущие должности, а также на менеджерские позиции, предпочитая растить таких специалистов сами. И это дает результат, карьерная лестница и в доле, и в компании работает достаточно хорошо. За последние четыре года из отдела тестирования выросли один директор по разработке и два вице-президента.

5 Конечно же, такие требования есть. Думаю, в той или иной мере они есть у любой компании. Наша специфика заключается в том, что наш софт предназначен для виртуализации и управления ИТ-инфраструктурой облачного провайдера, поэтому тестировщик обязан на хорошем уровне знать, что это за инфраструктура и как ею управляют. Для наших тестировщиков не составляет труда развернуть и настроить Microsoft Exchange или Lync, разобраться в дебрях конфигурации Apache или изолировать хитрую ошибку где-то в системных компонентах Linux. Поэтому часто на должность тестировщика мы ищем людей с серьезным опытом в системном администрировании или тех, кого принято называть «линуксоидами». На последнем собеседовании, например, у меня был сисадмин с восьмилетним опытом работы.

Кроме того, у нас необходимо знание английского языка, хотя бы на уровне pre-intermediate, поскольку весь внутренний документооборот ведется на английском.

Кирилл Фальк,

тест лид .NET-команды в JetBrains

1 Для нас самое важное, чтобы тестировщик любил свою работу, любил тестируемый продукт, глубоко и широко понимал его возможности, какие проблемы он позволяет решить в боевой обстановке у пользователей. Без понимания тестируемого продукта ничего не получится.

Так как мы живем в мире .NET, то тестировщик также должен понимать, как в этом мире все устроено, что такое CLR, garbage collector. В линейке .NET-продуктов есть разные (ReSharper, dotTrace Performance, dotTrace Memory, dotCover). Ведущим продуктом в .NET-департаменте у нас является ReSharper, соответственно человек, попадающий в команду тестирования ReSharper, должен понимать, как устроены IDE (в частности, Microsoft Visual Studio), иметь хотя бы общее представление о различных .NET-технологиях, языках программирования. С другой стороны, если человек – профессионал в своей области, то изучить новую технологию или новый подход для него не проблема.

Соответственно мы также рассматриваем людей с опытом тестирования Java/веб-ориентированных проектов.

Тестировать люди могут по-разному, иметь разный опыт (ручное/автоматизированное тестирование/performance testing), но мы в первую очередь смотрим на то, как тестировщик думает, рассуждает, готов ли он к самообучению, может ли сам ставить себе задачи и успешно их решать, насколько комфортно он чувствует себя при работе в команде.

Заходите, выбирайте, покупайте!

Вам нравится наш журнал? Вы можете с ним не разлучаться!

Что для этого нужно сделать?

В нашем интернет-магазине по адресу: <http://samag.ru/catalogue> можно приобрести отдельные номера журналов «Системный администратор» и «Бит» и книги наших авторов.

Также там можно купить приятный глазу и душе и одновременно полезный в хозяйстве сувенир с фирменной символикой «Системного администратора»! Настольные игры «Локалка» и «Outsourcer», кружку «Солнечное настроение», футболки «Системный администратор» и «Программист», пятнашки-антистресс «Собери мозги» и многое другое.

Самовывоз (Москва, Шереметьевская улица, дом 85, строение 2) или доставка Почтой России. Стоимость доставки 200 р.



E-mail: sa@samag.ru Tel.: (499) 277-12-41 Fax: (499) 277-12-45

2 Главное – это голова. А инструментарий у нас разный – начиная от всей линейки MS Visual Studio, различные VCS, и заканчивая тулзами для изучения кеш-дампов, например, WinDbg. Используем, конечно, свои же тулзы для профилировки памяти и производительности программы.

3 Особых требований у нас нет. Мы обращаем внимание на то, что человек умеет и чему обучился. На наличие диплома (красный/синий) мы смотрим в последнюю очередь.

4 Повторюсь, нам важно, что человек знает и что умеет. Но мы стараемся набирать людей уже с опытом тестирования больших коммерческих продуктов.

5 Большинство наших пользователей – программисты, и соответственно тестировать все продукты надо стараться с позиции программиста. Если говорить, например, о ReSharper, одну и ту же задачу разные программисты могут реализовать по-разному, соответственно нам это надо учитывать при тестировании функциональности.

Олег Строкаты,

руководитель отдела тестирования компании «1С-Битрикс»

Мы работаем в основном в области веб-технологий, поэтому рекомендации даются с этой поправкой.

1 Необходимо понимание специфики среды и объектов тестирования. Если говорить о веб-разработке, то нужны знания работы веб-технологий, схемы работы веб-сайтов, веб-серверов, баз данных, умение их конфигурировать, хотя бы на базовом уровне.

Знание английского языка на уровне выше начального, а лучше среднего, что позволит читать различные мануалы.

Наличие грамотного русского (или английского) языка и умения также грамотно и понятно излагать свои мысли. Это позволит быстрее донести до разработчика суть найденной проблемы и прийти к ее решению.

Необходимо обладать фантазией, чтобы сгенерировать максимально разумное количество прецедентов работы с продуктом.

Мы ведь говорим о квалифицированном тестировщике. Такой тестировщик должен уметь работать с инструментами автоматизации, то есть должен уметь программировать на уровне «выше базового», а лучше «хорошо». Это позволит разработать различные фреймворки, скрипты и инструменты автоматизированного тестирования, которые в итоге помогут сэкономить много времени и избежать рутинного труда.

2 Система трекинга ошибок. Есть как платные, так и бесплатные, например, Mantis, Redmine, Bontq. Установленные популярные браузеры с необходимыми версиями с консолью веб-разработчика. Сегодня это Google Chrome последней версии, Firefox последней версии, Internet Explorer 8 и 11, Safari последней версии. Утилиты для быстрого снятия скриншотов с возможностью рисования на них. Например, бесплатный FastStone или Joxi. Инструменты автоматизации. Их набор определяется спецификой разработки автотестов. Это могут быть установленные фреймворки и библиотеки. Например, установленный Eclipse или MS Visual Studio с подключенными би-

блиотеками Selenium. Если разработка автотестов ведется серьезно и долго, то не лишней будет система контроля версий проекта по автоматизации.

3 К сожалению, уровень образования и знаний большинства современных выпускников вузов и средних специальных учреждений не дает никакой возможности полагаться на имеющийся у них диплом. Мы не предъявляем особых требований к образованию. Решающее значение имеют опыт и навыки.

4 Опыт работы в тестировании необходим. Это более говорящий показатель, нежели наличие диплома. Желателен опыт работы в профильной среде от года.

5 Особых специфических требований нет. Как наиболее важное я бы выделил наличие достаточной самостоятельности у кандидатов, способности довести проблему до ее решения, а также умение работать быстро.

Юрий Трухин,

эксперт по облачным технологиям InfoboxCloud

1 Хороший тестировщик должен уметь понимать устройство сложных систем. Должен обладать навыками общения с коллегами. Быть устойчивым к рутинной работе и справляться с резко возрастающей нагрузкой в работе. Должен обладать не только навыками автоматизации тестирования, но и иметь и реализовывать новые идеи в сфере тестирования ПО, проявляющиеся в разработке ПО для тестирования, оптимизированного под конкретную задачу. Тестировщик всегда и разработчик.

2 Jenkins/Teamcity, базовые инструменты UNIX-систем, знание методик всех видов тестирования и умение аргументировать выбор инструмента в каждом из видов. Инструментов много, например, JUnit, WebDriver, Maven, Thucydides и др.

3 Выполняется проверка знаний в выбранной области до начала работы. Если вы обладаете необходимыми, у вас большой шанс. Плюс – участие в открытых проектах, где можно увидеть ваш реальный вклад в работу. Вы должны обладать умением объективно оценивать трудоемкость задач и координировать команду тестировщиков.

4 Плюсом является опыт программирования на Python, Java, PHP, C++ (умение читать и понимать код, лучше – опыт промышленной разработки). Необходимо знание основ ООП. Важнее умение тестировать, а не красивая запись в трудовой книжке. Минимум год работы по специальности – необходимое условие. Нужен опыт работы с UNIX (Linux, FreeBSD).

5 Мы стремимся максимально эффективно использовать человеческие ресурсы, поэтому автоматизация всего, что возможно, очень важное качество. Из-за специфики деятельности компании сфера работы тестировщика может пересекаться с программистской, сферой сетевого администрирования и оценки качества работы больших систем. Важна высокая внутренняя мотивация. Каждый сотрудник должен быть безразличным к качеству работы всей системы в целом и внимательным к мелочам, не ограничиваясь кругом ежедневной работы. **БОГ**

Подготовил Игорь Штомпель

Визитка

ВАЛЕРИЯ ЧЕРНЕЦОВА, руководитель отдела аналитики
рекрутингового портала Superjob.ru



От Москвы до Бреста не прожить без тестов

Исследовательский центр рекрутингового портала Superjob.ru подготовил для читателей журнала «Системный администратор» обзор предложений и требований работодателей, ищущих тестировщиков ПО

Регион	Диапазон I Без опыта работы на данной позиции	Диапазон II С опытом работы от 1 года	Диапазон III С опытом работы от 2 лет	Диапазон IV С опытом работы от 3 лет
Москва	30 000 – 40 000	40 000 – 50 000	50 000 – 85 000	85 000 – 140 000
Санкт-Петербург	24 000 – 33 000	33 000 – 42 000	42 000 – 68 000	68 000 – 100 000
Красноярск	18 000 – 24 000	24 000 – 30 000	30 000 – 50 000	50 000 – 85 000
Екатеринбург	20 000 – 27 000	27 000 – 34 000	34 000 – 57 000	57 000 – 95 000
Владивосток	21 000 – 30 000	30 000 – 36 000	36 000 – 60 000	60 000 – 110 000

Типичный функционал

- > разработка тест-стратегии, тест-планов, тест-сценариев;
- > проведение различных типов тестирования согласно плану;
- > регистрация и мониторинг обнаруженных дефектов;
- > написание отчетов о проведенных тестированиях;
- > взаимодействие с разработчиками и аналитиками.

Портрет соискателя

- > 29 лет – средний возраст
- > 63% – мужчины
- > 67% – имеют высшее образование
- > 2 года – средний срок работы на последнем месте
- > 18% – готовы к переезду
- > 57% – на момент размещения резюме не имели работы

Зарплатный диапазон	Требования и пожелания к профессиональным навыкам
Диапазон I	> Неполное высшее образование (техническое/ИТ) > Уверенный пользователь ПК > Знание основ администрирования Windows/*nix-систем > Понимание общих принципов разработки ПО > Базовое знание одного или нескольких языков программирования (C++, Java, Delphi) > Знание стандартов и методик тестирования > Знание английского языка на уровне чтения технической документации
Диапазон II	> Понимание принципов работы реляционных баз данных > Знание SQL на уровне написания простых запросов > Опыт тестирования ПО от полугодя Возможные пожелания: > знание технологий и средств автоматизации тестирования > знание скриптовых языков > опыт в администрировании Windows/*nix-систем
Диапазон III	> Высшее образование (техническое/ИТ) > Знание систем автоматизации тестирования > Навыки документирования процесса тестирования и обнаруженных дефектов > Навыки составления планов тестирования > Опыт проведения различных видов ручного и автоматического тестирования (системное, функциональное, регрессионное, тестирование производительности и т.д.) > Опыт разработки тестовой документации (test-plans, test-cases, use-cases) > Опыт использования систем баг-трекинга > Опыт работы тестировщиком ПО от 1 года Возможные пожелания: > навыки программирования/разработки и внедрения ПО > опыт работы с системами менеджмента тестовой документации
Диапазон IV	> Уверенное знание нескольких языков программирования > Опыт тестирования сложных приложений > Опыт работы тестировщиком от 3 лет Возможные пожелания: > знание английского языка на разговорном или свободном уровне > опыт руководящей работы

SuperJob



Визитка

ВЛАДИМИР ГАКОВ, журналист, писатель-фантаст, лектор. Окончил физфак МГУ. Работал в НИИ. С 1984 г. на творческой работе. В 1990-1991 гг. — Associate Professor, Central Michigan University. С 2003 г. преподает в Академии народного хозяйства. Автор 8 книг и более 1000 публикаций

Прямой путь к богатству

Секрет простой — напрямую, без посредников. Непокоримое следование этому и другим собственным бизнес-принципам позволило Майклу Деллу вывести свою компанию в мировые лидеры, а самому войти в первую двадцатку самых богатых людей планеты по версии журнала *Forbes*

Причем до появления в знаменитых рейтингах вышеупомянутого журнала другого «вундеркинда» Марка Цукерберга Делл оставался самым молодым из супербогачей: в его возрасте столько денег не имел даже сам «великий и ужасный» Билл Гейтс.

Деловой мальчуган

О том, как Делл превратил стартовую \$1000 в 50 млрд, ходят такие же легенды, как о «гаражах», в которых собирали первые модели Apple и Hewlett Packard, или о чертеже первого портативного Compaq, нарисованного на ресторанной салфетке.

К бизнесу Майкл Сол Делл приобщился с малолетства, и первый толчок дали, сами того не подозревая, родители. Отец будущего миллиардера работал стоматологом, а мать — финансовым советником. Это была процветающая еврейская семья, хорошо известная в деловых кругах Хьюстона. Когда их сынишке стукнуло пять, на дворе стоял 1970 год, в стране разразился нефтяной кризис, и вся нация с изумлением вспоминала о таком полузабытом в Америке феномене, как очереди — на сей раз у станций заправки. Поэтому в доме Деллов разговор постоянно шел о деньгах. Родители полагали, что вряд ли маленький сын прислушивается к таким скучным взрослым беседам про индекс инфляции, учетные ставки, розничные цены и тому подобное.

Однако сынок-вундеркинд слушал и внимательно впитывал услышанное. В семилетнем возрасте он купил себе первый калькулятор. А годом позже второклассник Делл начал свой первый бизнес — наладил на своей улице розничную торговлю леденцами. От клиентов — соседских ребятишек — не было отбоя.

Перейдя в следующий класс, он наткнулся в одном из журналов на отрывной талон-приглашение на учебу в одном из техасских колледжей. С помощью таких «приемных талонов» многие платные учебные заведения США вербуют себе будущих абитуриентов, заканчивающих среднюю школу. Делл тоже решил попробовать и выслал заполненную открытку по указанному адресу. Каково же было его изумление, когда спустя несколько недель к нему явилась университетская

«вербовщица» с подробной анкетой и сведениями для поступления в свой вуз. Но еще больше изумилась гостья, увидав перед собой «потенциального студента» — третьеклассника в махровом халате и с мокрой после ванны головой. «Дамочка потеряла дар речи, — вспоминал Делл, — зато я открыл для себя истину: необязательно знать клиента в лицо, чтобы заставить покупать твой товар».

В старших классах Делл наладил уже вполне серьезный бизнес — самодельную филателистическую биржу, заработав на первые марки мытьем посуды в китайском ресторане. Однако в отличие от миллионов других юных филателистов Делл задался вопросом: зачем переплачивать посредникам — продавцам марок, когда можно напрямую связываться с филателистами, желающими что-то продать и купить?

Он упрямился сверстников дать ему свои марки на, как бы сказали сегодня, «консигнацию». А затем поместил объявление о продаже марок от имени некоей фирмы Dell's Stamps в филателистическом журнале, начувая текст одним пальцем на отцовской пишущей машинке. Шутя заработав \$2000, юный коммерсант усвоил свой первый и главный урок в бизнесе, ставший основой его последующего невероятного взлета, — по возможности избегать посредников.

Фарцовка вместо лекций

На доходы, полученные с торговли марками, Делл купил себе первый компьютер Apple II — в то время единственную модель того, что позже назовут «персоналкой». Любопытный и предприимчивый подросток первым делом разобрал покупку до винтика, чуть не доведя родителей до инфаркта — они подумали, что сынок разломал дорогостоящую покупку! Как и для всех подобных юных гениев-технарей, разборка имела единственную цель — посмотреть, как там все устроено внутри. Разобравшись в компьютерных внутренностях, Делл разработал алгоритм апгрейда, после чего наладил торговлю модернизированными компьютерами среди одноклассников.

В 16-летнем возрасте он смог заработать уже не на один компьютер. Летом Делл устроился на подработку в редакцию газеты Houston Post, где молодого «сдельщика» бросили на самое безнадежное (лето, отпуска...) дело — искать новых

подписчиков по телефону. Однако прирожденный бизнесмен быстро оптимизировал свой труд – вместо того, чтобы наугад обзванивать всех подряд, он в результате блиц-опроса выяснил, что новых клиентов перспективнее всего искать в двух социальных группах – среди молодоженов и тех, кто недавно переехал в Хьюстон, купив квартиру в рассрочку. Следующий шаг напрашивался сам собой – раздобыть на почте адреса всех, кто в последнее время получал по почте свидетельство о браке или банковские извещения по оплате за ипотеку, и им-то в первую очередь выслать подписные квитанции.

Благодаря смекалке Делла редакция за месяц обзавелась тысячами новых подписчиков, а сам он заработал огромные для его возраста деньги – \$18 000. Больше, чем зарабатывали за год его школьные учителя! На первый в его жизни серьезный гонорар, как вспоминал сам Делл, были куплены «автомобиль, стереосистема и все прочее, что составляет счастье 16-летнего подростка. Но самое главное – компьютеры и комплектующие. Бизнес не должен был простаивать!»

Поступив в Университет штата Техас в Остине, Делл практически весь первый курс просидел не в аудитории, а в «общаге», где была налажена бойкая сборка и торговля компьютерным железом. Он быстро разглядел слабые стороны тогдашней торговли компьютерами – дилеры не затруднялись поиском нестандартных решений, а просто закупали готовый товар у производителей или дистрибьюторов и перепродавали его с наценкой. «Если бы мы заглянули внутрь персонала ценой \$2000, то обнаружили бы там начинки в лучшем случае долларов на шестьсот!» – вспоминал Делл. О сроках поставки и говорить нечего – доставка компьютера затягивалась на многие месяцы, иначе говоря, покупатель получал отнюдь не последнее слово техники, которая сама прогрессировала не по дням, а по часам.

Оживленная «торговая точка» в общежитии не вызвала энтузиазма у преподавателей и руководства университета. И когда Делл решил завязать с учебой и покинуть alma mater без диплома, никто его не отговаривал. В мае 1984 года, собрав стартовую сумму в \$1000, он зарегистрировал собственную фирму – Dell Computer Corporation. Как пишет один из его биографов, «на эту тысячу баксов можно было: а) закатить пьянку, которой мир не знал со времен победы Американской революции; б) купить подержанный «Форд»; в) создать фирму по продаже компьютерного железа. Делл выбрал вариант с».

Из рук в руки

Однако торговать компьютерами в кампусе, где все всех знают, и по всей стране – вещи несравнимые. Новоявленному бизнесмену пришлось первым делом задуматься: как достучаться до сердца и, главное, кошелька, массового покупателя, и выяснить, какой именно компьютер ему нужен? Тут очень пригодился «телефонный» опыт маркетинга – половину столов в снятом первом офисе занимали телефонистки, собиравшие заказы, а на второй половине столов по этим заказам собирали компьютеры. Штат новоиспеченной компании составляли упомянутые телефонистки, упаковщики готовой продукции и «трое парней с отвертками, которые, собственно, и были «производством». Эта «веселая компания» уже в первые годы производила продукции на \$50-80 млн.



Майкл Делл: «Клиент всегда любим и счастлив – благодаря нам»

В голове Делла сформировалась схема его «фирменно-го» бизнеса – продажа компьютера, собранного по заданной спецификации будущего покупателя! На тогдашнем компьютерном рынке эта бизнес-модель была революционной – компания приступала к сборке компьютера только после поступления заказа от клиента. Торговые затраты соответственно снижались (любая крупная компания неизбежно сталкивается с большим процентом возврата – нераспроданной или морально устаревшей продукции), соответственно снижалась и цена. Важнейшим элементом в схеме Делла стал прямой контакт с потребителем, что обеспечивало ценнейшую и, главное, бесплатную информацию относительно векторов покупательских привычек и предпочтений.

В своей книге-бестселлере, названной «Напрямую от Делла» (1999), автор метода так описывает его суть: «С раннего возраста меня возбуждала идея исключения любых промежуточных шагов и всех и всяческих посредников. Dell продает компьютеры напрямую, напрямую же связана с поставщиками, и общение сотрудников внутри компании происходит также напрямую. Никаких посредников – неэффективных и вовсе не столь необходимых, какими они обычно себя представляют. Мы называем это «моделью напрямую» (direct model), и она пронизывает нашу компанию снизу доверху».

«Раньше никому в голову не приходило спрашивать покупателя, что ему нужно, – продолжает Делл. – Продавцы просто доставляли коробки с полком и начинали нахваливать товар, который к тому времени успел порядком устареть».

Началось все с продажи по почте (mail-orders), но особенно заиграла эта схема с наступлением эры Интернета – Делл одним из первых продемонстрировал практически неограниченный коммерческий ресурс Всемирной сети. Корпоративный сайт www.dell.com пользовался бешеной популярностью – на нем можно было не только собрать необходимую конфигурацию, но и воспользоваться корпоративной базой и сервисным обслуживанием – также в режиме онлайн. Ежедневный объем продаж в сети уже спустя два года превысил \$1 млн. С появлением в бизнес-арсенале компании Интернета Dell Computer уже не нужно было, как остальным игрокам на рынке, ожидать конца квартала, чтобы получить и проана-

лизировать суммарный заказ от дилеров. Отдел маркетинга получал информацию о клиентах буквально ежеминутно и был в курсе того, что им на сегодняшний день нужно.

Любовь к клиенту Деллу не помеха

«Мы очень старались, – писал Делл, – и даже редкие ошибки (а кто их избегал?) обращали в свою пользу. Потому что никогда не позволяли себе поскользнуться дважды на одном месте – придиричиво «выслеживали» собственные просчеты, анализировали ошибки и делали необходимые выводы. Но и ошибаясь, мы неизменно хранили верность нашим основным корпоративным принципам».

Главным из них стала на первый взгляд очевидная аксиома торговли: клиент всегда прав. Однако в Dell Computer был произведен вроде бы незначительный, но оказавшийся решающим сдвиг в оценках: «клиент всегда любим и счастлив – придиричиво». Как писал автор принципа, «эту философию легко сформулировать, но оказывается, что ей легко и следовать. Просто нужно помнить о наших главных ценностях постоянно – каждый день. Нужно делать все и даже больше, чтобы наш клиент был счастлив. Это вовсе не означает, что мы готовы предоставить клиенту все, чего его душа пожелает. Прежде всего необходимо увидеть и понять, за что и сколько именно клиент готов заплатить... А уж наша задача – предоставить ему весь спектр услуг, включая необычные – например, золотые, серебряные и платиновые дисконтные карты для постоянных клиентов».

По мере роста компании Деллу пришлось заняться и выстраиванием управленческой структуры, которая бы идеально подходила к созданной им бизнес-философии. Так он пришел к идее компании, в которой, несмотря на размеры, отсутствует бич большинства крупных организаций – большое число «уровней ответственности», что обычно приводит к прогрессирующей бюрократизации. Делл настаивал, что ему не нужны сотрудники, которые знают о наличии той или иной проблемы, но всячески отлынивают от принятия решений, предпочитая перекладывать его на других: «Проблема фирмы – это и твоя проблема, тебе ее и решать».

Не обходилось без накладок, но Делл с самого начала ввел для своего менеджмента негласное правило: никогда не ставить себе цель стопроцентно «безошибочных» решений (что сковывает творческую инициативу), но и не повторять однажды совершенных ошибок.

В начале 1990-х годов компания решила испытать фортуна на новом для себя оптовом дилерском рынке – по примеру модных тогда «компаний-складов» вроде CompUSA или Sam's Club. Обе компании продавали компьютеры

по частям, а покупатель приобретал необходимые детали, чтобы самому модернизировать свой компьютер. Однако очень быстро в Dell Computer признали, что решение было неудачным: отношения «склад – клиент» (заявка – доставка – оплата) разрушали годами создававшуюся схему Делла (заказ – создание товара под заказ – доставка – сервис). И руководство компании вернулось к «розничной» схеме. «Работая по «складской» схеме, – резюмировал негативный опыт Делл, – мы не могли сделать клиента счастливым. К счастью, мы вовремя свернули эту деятельность – без ущерба для своей репутации».

Принцип Domino

Хотя завистники и конкуренты обычно сравнивают корпорацию Делла с гигантской белой акулой – самым крупным, агрессивным и примитивным морским хищником, пожирающим всех, кого встретит на пути, – более точным представляется другое сравнение. С самой известной в США компанией по производству пиццы – Domino.

Любимое кушанье миллионов американцев в Штатах кто только не производит, но равных Domino Pizza нет. По крайней мере не было до определенного времени. Потому что, заказав у этой компании пиццу по телефону в любой точке страны, вы можете засекают время – если разность не доставит заказ в течение 20 минут, вы получите свою пиццу бесплатно. Пункты Domino Pizza расположены на каждом углу, а мальчики-разносчики на мотороллерах и велосипедах – особая гордость фирмы. Они способны просочиться сквозь дорожную пробку, выбрать оптимальный маршрут, а если время поджимает, то и «нарушить», рванув на красный свет или по тротуару.

В упомянутое выше «свое время» это привело к карательным мерам по отношению к компании – ей запретили упоминать в рекламе пресловутые «20 минут», дабы не провоцировать разносчиков на правонарушения. Однако поколения американцев привыкли к «20 минутам», и сегодня мало кто засекает время – шансы на получение «халвяной» пиццы минимальны.

Аналогом Domino Pizza в компьютерном мире стала фирма Dell Computer. С помощью пионерского метода прямого маркетинга и впервые же внедренной в 1986 году системы послепродажного сервиса и поддержки Делл превратил свою компанию в продавца компьютерных систем номер 1 в мире. Клиенты Dell были уверены – проблем с ремонтом, обслуживанием, комплектующими у них не будет. Достаточно только позвонить и – засесть время.

К середине 1990-х годов компания Майкла Делла благополучно справилась с неизбежными проблемами «переходного



Майкл Делл. Миллиардер и бизнес-философ

возраста». Для нее они оказались особенно болезненными, если учесть, какой невероятный, не знавший прецедентов взлет испытала Dell Computer за неполные десять лет. Однако глава компании справился с трудностями своим методом – закрывая все промежуточное, неэффективное и малоперспективное. Но закрывая не насовсем, не отсекая, а лишь замораживая на время, переводя «в тыл» – Делл хорошо усвоил, что малоперспективное сегодня может оказаться «горячим пирожком» завтра, поэтому важно держать свои резервы наготове.

Показательным в этом смысле представляется временный уход Dell с рынка ноутбуков. Он удивил многих (этот сектор рынка в последние четверть века развивался бурно и стабильно – до появления смартфонов), однако Майкл Делл, говоря военным языком, лишь совершил «тактический отход на заранее подготовленные позиции для выравнивания фронта и перегруппировки сил перед решающим наступлением». Оно не замедлило последовать: вскоре на рынке с триумфом дебютировала новая линия ноутбуков Dell – Latitude, которые били конкурентов за счет своих «долгоиграющих» батареек, которым в ту пору не было равных.

Примерно тогда же компания Dell вместо того, чтобы продолжать сражение за увеличение своей доли на рынке настольных и портативных PC, частично переключилась на новое и, казалось, не сулившее больших выгод дело – производство серверов. И в 1996 году вызвала новую сенсацию своими серверами PowerEdge. Они оказались столь удачными (и, как всегда у Dell, более дешевыми по сравнению с аналогичными моделями конкурентов), что за неполные два года компания переместилась на мировом рынке серверов с десятого места на второе, увеличив свою долю почти вдвое.

Иногда они возвращаются

«Моя компания – это наглядное доказательство того, что можно увидеть и извлечь прибыли из таких возможностей, которые ваши конкуренты просто отказываются видеть, считая их неосуществимыми. Для того чтобы думать не так, как все, совсем не обязательно быть гением или провидцем, или даже дипломированным специалистом. Все, что требуется, – это мечта, а еще жесткая структура, которая будет служить рамками для осуществления этой мечты». Автор процитированного пассажа в одном из своих интервью скромно заметил, что с удовольствием поужинал бы в компании единомышленников – Альберта Эйнштейна, Исаака Ньютона, Энди Гроува и Билла Гейтса.

Впрочем, скромничать Деллу не пристало. В 2002 году его компания продала по почте – обычной и электронной – компьютеров на \$35 млрд и обеспечила работой 36 тысяч человек. Причем все это на фоне тотального падения продаж в компьютерной отрасли! В следующем году доля Dell Computer на рынке составила 22%, а сама компания по итогам года заняла 33-е место в списке 500 крупнейших компаний мира. В первые годы нового века и тысячелетия на долю дитища Майкла Делла, сменившего название на Dell Inc., приходилось около 13% мирового рынка персональных компьютеров, что позволило компании стать единоличным лидером, обогнав главного на то время конкурента – Compaq.



Dell Inc. сегодня – одна из крупнейших высокотехнологичных компаний в мире

Может быть, в этом триумфе и заложена главная причина того, что десять лет назад, в 2003 году, 39-летний Делл добровольно уступил свой пост CEO ближайшему соратнику – Кевину Роллинзу, человеку номер 2 в компании. Создавалось впечатление, что Деллу тогда, как альпинисту, покорившему Эверест, просто больше нечего делать.

Покинув руководящий пост в своей компании, Делл переключился на инвестиционный бизнес. Еще в 1998 году он основал другую компанию – MSD Capital L.P. (название состоит из инициалов основателя). Этот второй бизнес Делла был также успешен, но все же спустя почти 10 лет совет директоров «первой» компании – Dell Inc. – упрямил ее создателя вернуться. И в последний день января 2007 года Майкл Делл снова уселся в кресло руководителя компании, которую сам создал двадцать с лишним лет назад. А в начале января прошлого года мир бизнеса потрясла сенсационная новость: Делл, имевший более 12% акций, предложил акционерам Dell Inc. продать ему все остальные – за \$24,4 млрд. Сумму рекордную со времен Великой депрессии!

Впрочем, Майкл Делл прекрасно отдает себе отчет, за что он готов выложить такие деньги. Dell Inc. сегодня – одна из крупнейших высокотехнологичных компаний в мире. Обладая штатом в более чем 100 тысяч человек, компания продает широкий ассортимент товаров на рынке high tech: компьютеры, серверы, средства хранения информации, сетевые устройства, программное оборудование, периферию, телевизоры высокого разрешения, камеры, принтеры, MP3-плееры, ИТ-сервисы, а также разнообразную бытовую электронику сторонних производителей. Ныне Dell Inc. открывает шестую десятку в списке 500 крупнейших компаний мира по версии журнала Fortune, в прошлом году занимая третье место по продаже «персоналок» (после Lenovo и HP) и первое по продаже мониторов к «персоналкам».

Да и сам единоличный владелец этого многомиллиардного бизнеса входит в первую полусотню не менее знаменитого «списка Forbes», чья «персональная капитализация» в позапрошлом году была оценена этим журналом в \$16 млрд. Так что, перефразируя известный лозунг еще советских времен, можно заключить: дело Делла живет и побеждает! **ВОВ**

Майкл Делл



Основы бизнеса Майкл Делл (родился в 1965 году) – основатель, CEO, а ныне и единоличный владелец «одно-семейной» компании (см. стр. 92-95) – начал постигать в восемь лет, когда торговал леденцами среди сверстников. Потом он переключился на почтовые

марки, которые продавал уже солидным взрослым людям (не подозревавшим, что продающая марки «фирма» состоит из одного подростка-третьеклассника). И, наконец, принялся за компьютеры. Да так удачно, что стал самым молодым – на то время – долларовым миллиардером, а его компания – одним из крупнейших торговцев компьютерной техникой в мире.

Самое замечательное в нашем бизнесе то, что благодаря связи с клиентами мы имеем возможность мгновенно получать от них интересующую нас информацию.

Предметом моей особой гордости является то, что, даже когда мы делали ошибки, мы никогда не отступали от наших основных принципов.

Клиент не может быть просто удовлетворен. Клиент должен быть доволен! Однако это вовсе не означает, что нужно каждому клиенту обеспечивать ВСЕ, что он хочет. Сначала нужно определить с тем, за что клиент готов заплатить. Если он говорит, что хотел бы, чтобы вы оказывали ему поддержку 24 часа в сутки семь дней в неделю, чтобы вы делали то-то и то-то, спросите его в ответ, а готов ли он за это платить, потому что все эти услуги стоят денег.

Как мне удается удерживаться на плаву все эти годы? Во-первых, я получаю от своей работы удовольствие; во-вторых, я думаю, что достиг успеха, потому что всегда задавал себе один и тот же вопрос: «Что еще я могу сделать, чтобы улучшить работу компании?»

Каждый должен быть нацелен на успех компании, проблема фирмы – твоя проблема, и тебе ее решать. Организация имеет такую хорошо заточенную культуру, что, если кто-то не шевелится и не нацеливает себя на общий результат, пробует разводить бюрократию, организация обходит его. Нам не нужны такие работники.

В нашей компании никто не имеет права сказать, что только потому, что он проработал у нас более 10 лет, он может почивать на лаврах и не стремиться к дальнейшему развитию.

Чтобы мотивировать сотрудника, заставить его почувствовать себя совладельцем компании, нужно подходить к сотрудникам с теми мерками, которые он в состоянии воспринять.

Изначально мы создали структуру управления, в которой нет и не должно быть много уровней. Такова модель компании, ориентированная на результат, активной и агрессивной. Эта система должна быть саморитичной, поощряя желание работника открыто говорить о проблемах в фирме. Если кто-то, зная о существовании проблемы, пытается сделать вид, что ничего не происходит, – такой человек нам не нужен.

Так называемые милые парни обычно заканчивают дистанцию последними.

Есть множество вещей, которые ведут к успеху. Я не люблю заниматься только тем, чем мне нравится заниматься, и поэтому делаю не то, что нравится, а то, что пойдет на пользу делу.

Да, наш бизнес вертится вокруг технологий. Но не только – нас равным образом волнуют производственные операции и отношения с покупателями.

Двадцать лет и 40 миллиардов долларов... – кажется, это неплохие круглые числа!

Я думаю, что в век Интернета обязательно победит модель развития компании, основанная на открытом сотрудничестве, а не та, что основана на распространении права собственности на все и вся, как у многих наших конкурентов... Подход, основанный на правах собственности на те или иные технологии и на «вертикальных» сервисных решениях, устарел. Во всяком случае, отвергается большинством покупателей во всем мире.

Идея «быть всем для всех» – это идея из прошлого.

Единственное, что развивается быстрее Linux, – это Linux на компьютерах Dell.

Покупатели – вот кто привел Dell к успеху в первую очередь. И если мы окажемся достаточно мудрыми и проворными, чтобы вовремя услышать нужды покупателей, мы победим.

Продай пользователю не то, что завалилось на складе, а то, что ему нужно, и дешевле, чем продают другие.

Для того чтобы добиться успеха, не обязательно быть гением или провидцем. Не обязательно даже заканчивать колледж. Достаточно просто иметь мечту и удачно ее структурировать. **БОР**

Владимир Гакос

Системный администратор

Издается с 2002 года

Включен в Российский индекс научного цитирования www.elibrary.ru

Генеральный директор

Владимир Положевец

Главный редактор

Галина Положевец, chief@samag.ru

Исполнительный директор

Владимир Лукин, maker@samag.ru

Заместитель главного редактора

Ирина Ложкина, lozhkina@samag.ru

Заместитель главного редактора,

официальный представитель редакции в Украине

Сергей Яремчук, grinder@samag.ru

Главный бухгалтер

Надежда Кан, buch@samag.ru

Главный редактор приложения «Open Source»

Дмитрий Шурупов, osa@samag.ru

Реклама

Лариса Говорченко, govorchenko@samag.ru

Распространение

Олег Иванов, subscribe@samag.ru

Юридический отдел

Владимир Столяров, stolyarov@samag.ru

Дизайн обложки

Михаил Лебедев

Иллюстрация

Виктор Чумачев

Дизайн-макет

Марина Рязанцева,

Дмитрий Бессонов

Редакционная коллегия

Д. Ю. Гудзенко, кандидат технических наук, директор Центра компьютерного обучения – «Специалист» при МГТУ им. Н.Э.Баумана

Д. Ю. Диндич, доктор технических наук, ведущий преподаватель Центра компьютерного обучения «Специалист» при МГТУ им. Н.Э.Баумана

А. С. Крюковский, доктор физико-математических наук, профессор, лауреат Государственной премии СССР, декан факультета информационных систем и компьютерных технологий Российского нового университета

Э. С. Клышников, кандидат технических наук, доцент, зам. декана по научной работе факультета информационных технологий и вычислительной техники МИЭМ НИУ «Высшая школа экономики»

В. Е. Синицын, кандидат физико-математических наук, доцент Уральского федерального университета

Д. В. Силаков, кандидат физико-математических наук, старший архитектор российского Центра разработки решений на основе СПО «POCA»

С. Р. Тумковский, доктор технических наук, профессор, зав. кафедрой факультета информационных технологий и вычислительной техники МИЭМ НИУ «Высшая школа экономики»

А. В. Тетюшев, кандидат технических наук, доцент Вологодского государственного технического университета

Экспертный совет

Радий Ачилов, главный специалист по защите информации

Сергей Барамба, эксперт по системным решениям

Алексей Бережной, эксперт по администрированию и ИБ

Андрей Бирюков, ведущий системный инженер по ИБ

Алексей Воронин, эксперт по вопросам разработки ПО

Константин Кондаков, старший директор по ИТ

Иван Коробко, эксперт по автоматизации процессов в Windows-доменах

Кирилл Сухов, ведущий специалист направления интернет-разработки

Леонид Шапиро, эксперт по ИБ и инфраструктурным проектам

Сергей Яремчук, эксперт по ИБ

Издатель

ООО «Синдикат 13»

Адрес редакции

129075, г. Москва, Шереметьевская ул., д. 85, стр. 2, офис 405,

тел.: (499) 277-12-41, факс: (499) 277-12-45

Сайт журнала: www.samag.ru

Отпечатано в типографии

ООО «Периодика» Тираж 17000 экз.

Все права на материалы принадлежат журналу «Системный администратор». Перепечатка и использование материалов в любой форме, в том числе и в электронных СМИ, без разрешения запрещены. При использовании материалов ссылка на журнал «Системный администратор» обязательна. Материалы, отмеченные знаком **BOF** публикуются на коммерческой основе. Редакция не несет ответственности за достоверность информации в материалах, опубликованных на правах рекламы.

ПРОБЛЕМ НЕТ – ЕСТЬ ЗАДАЧИ

Все говорят, что нужно делать, а мы помогаем достичь результата.
Решений много – найдите свое в «Системном администраторе» и «БИТЕ».

Зайдите на magzter.com, скачайте номер журнала на компьютер, планшетник, смартфон.
Читайте «Системный администратор» и «БИТ». Теперь это просто!

Android

iPhone

Windows Phone 8



iPad

Android Tablet



Системный администратор

БИТ



MAGZTER
DIGITAL NEWSSTAND
TAP • READ • ENJOY



Реклама



ЛЮБОЙ ДИСТРИБУТИВ СЕДЬМОЙ ПЛАТФОРМЫ НА ФЛЭШ-НОСИТЕЛЕ

*Всегда свежий!**



Альт Линукс 7.0 Кентавр

Альт Линукс  Десктоп 7.0



АЛЬТ ЛИНУКС 7.0
ШКОЛЬНЫЙ



Simply Linux 7.0

Интернет-магазин Альт Линукс: **shop.altlinux.ru**

*На флэш-носитель записывается актуальный стабильный релиз дистрибутива.